

# Commodore

# WORLD

Nº 14 - ABRIL 1985

315 PTAS.

## Sonido y música

**Serenata SID**

**Gráficos de alta  
resolución**

**Volando  
con el VIC**

**Telegrafía  
con ordenador**



## RITEMAN PLUS



## RITEMAN BLUE PLUS



RIT

### RITEMAN: PERSONAL/BUSINESS PRINTER

NUOVA GENERAZIONE DI IMPRESORAS PROFESIONALES. PERSONALES Y DE OFICINA CON MAXIMAS PRESTACIONES Y REDUCIDO TAMAÑO. A PRECIO CONTENIDO.

- MAYORES PRESTACIONES
- MENOR TAMAÑO
- MEJOR PRECIO

- Paralelo centronics, opcion RS232
- Velocidad 120, 140, 160 cps.
- Bidireccional optimizada.
- Tracción y fricción.
- Subrayado y retroceso.
- Sub y suprarrandices manual altura.
- Control por software y wordstar.
- Estalizador, replicado, física, etc.
- Impresión a 5, 6, 8, 10, 12 y 17 cpl.

RITEMAN

- ★ RITEMAN PLUS: 80 columnas, velocidad 120 cps. 91.000 ptas.
- ★ RITEMAN BLUE PLUS: Compatible IBM, 140 cps. 85.000 ptas.
- ★ RITEMAN V: Profesional, Buffer 2K, 180 cps. 93.000 ptas.
- ★ RITEMAN 15: 136 columnas, velocidad 160 cps. 155.000 ptas.
- ★ RITEMAN 15-IBM: 128 columnas, 160 cps. 156.000 ptas.

DE VENTA EN LOS MEJORES ESTABLECIMIENTOS ESPECIALIZADOS.

**DATAMON**

DATAMON, S. A.

PROYECTA: 285-MI, S.C. 1.ª  
TELEFONO: (00) 891 21 84

RESERVACION EN DINER 91

**RITEMAN:**

IMPRESORAS PROFESIONALES.

08001 - BARCELONA

RITEMAN



RITEMAN 15



# SUPER-INTERESANTISIMO

## PIRATERIA DE SOFT

—Comienza la polémica—

Con el número en la imprenta y ya cerrado, hemos recibido dos cartas, una de A.J.O.G. (pide no se publiquen sus datos) de San Sebastián y de José Emilio Bases Muñoz de Zaragoza sobre el editorial publicado en nuestro número 15 del mes de marzo, dedicado a la piratería de Soft.

Consideramos que estas cartas son de suficiente interés para publicárlas en su totalidad y comentarlas debidamente, por lo que les dedicaremos el espacio necesario en nuestro próximo número de mayo.

### Commodore World en discos

[Ya] Desde este número 14 del mes de abril, como ya anunciáramos en febrero, a partir de hoy editamos la revista en disco, paralelamente a la edición impresa. Recordamos que el disco irá en su estuche con su portada correspondiente a todo color.

El precio de cada disco es de 2.600 ptas. y 1.750 ptas. para los suscriptores.

La suscripción anual de 11 números es

de 3.750 ptas. La forma de pago es solamente por cheque o giro, y encartado en la revista tenéis el boleto de pedido.

### Tapas autocuendernables

Las tapas —Volumen 1— para doce números de Commodore World estarán ya disponibles a partir del 28 de abril. Estas tapas son integradas y no necesitan encuadernación posterior, ya que interiormente llevan unas pestañas para insertar directamente las revistas por su página central. Su precio es de 450 ptas., la forma de pago es, solamente, por cheque o giro y encartado en la revista tenéis el boleto de pedido.

### Números agotados ofrecidos

1, 3, 4, 5, 6 y especial 100 programas.

### RAMY Y ROMO

Como veréis, en este número, por razones de espacio nos vemos obligados a quitar el comic que habitualmente apa-

rece en nuestras páginas dirigidos a los peques —de dos a veinte años—. Esperamos que en el próximo número podamos disfrutar de la compañía de nuestros amigos.

### HACIA EL SUSCRIPTOR 8.000 RECORDATORIO

Sorteo entre TODOS los suscriptores de:

- 1) Viaje para dos personas, a elegir entre París, Lisboa, Atenas, Amsterdam, Niza, Viena, Cambrin, Mallorca.
- 2) Impresora donada por Microelectrónica y Control.
- 3) 80 programas valorados en 160.000 ptas. donados por la casa POKE, S.A. de Barcelona.

ULTIMO NUMERO DE SUSCRIPTOR = 7.881

## CONCURSOS PERMANENTES

### Colaboraciones de Programas y Magia

La verdad sea dicha: os están portando; las colaboraciones que nos están llegando son numerosas y de muy buena calidad. Por lo tanto, vamos a aclarar las condiciones de nuestros concursos permanentes.

- 1) Los sorteos se realizarán en los meses de julio y diciembre.
- 2) Se valorarán siempre entre premios, entre de 15.000, 10.000, 5.000 y 3.000 pesetas en metálico de "Commodore World" y cuatro del mismo valor en material de Microelectrónica y Control, entre todas las colaboraciones publicadas.
- 3) Se sorteará siempre tres premios, dos de 3.000, 2.000 y 1.000 pesetas en metálico de "Commodore World" y tres del mismo valor en material de Microelectrónica y Control, entre todas las contribuciones publicadas en la sección de "Magia".
- 4) Adicionalmente, se sortearán diversos premios de cintas, juegos, objetos curiosos, etc., entre todas las colaboraciones.
- 5) Los autores de las colaboraciones vendidas dentro de nuestro "Servicio de Cintas" percibirán el 20% del precio de la cinta.
- 6) Todas las colaboraciones deben venir escritas a máquina y los programas grabados en cinta (si es posible), o con el listado completo en impresora. Nuestras lecturas más juiciosas pueden recibir a mano pero con letra muy clara.
- 7) Quedan automáticamente descalificados tanto del sorteo como del "Servicio de Cintas" las colaboraciones que hayan sido enviadas a otras revistas.
- 8) Las colaboraciones se enviarán a "Commodore World", c/Barquillo, 21-3ª planta, 28004 Madrid.

### VIC 20

### TARJETA MULTIFUNCION

- Amplía 64 K RAM
  - Zócalo 8 K EPROM
  - ¡Traspasa y ejecuta desde cinta programas de cartucho! (Novedad)
- p.v.p.: 20.600 pts.

Pedidos a ICR  
C/ Farnés, 39  
Barcelona 08032

Elija forma de pago:

- ☐ Talón bancario
- ☐ Contrareembolso

## METEDURAS DE PATA

- En la sección de Magia, de nuestro número 11, en el programa titulado "números encolofonados", aparecía dos veces un cuadrado inverso entre comillas. Habría sido más correcto que era un "CLEAR".
- El número de teléfono de IDEALOGIC apareció equivocado en una de las reseñas, el número correcto es el (93) 330-33-68. Mil perdones al propietario del teléfono equivocado.
- En el listado del programa "reductor", las últimas líneas (de la 60000 en adelante) no tienen nada que ver con el programa, es una rutina que utilizamos nosotros para grabar las versiones de un programa en disco. Si alguno quiere utilizarla, sólo tiene que colocar el nombre del programa en la línea 60005 y hacer RUN60000 cada vez que quiera salvarlo. En el disco hay siempre dos versiones del programa, la "Backup" es la más antigua y la "last" la última. "No hay mal que por bien no venga".

## Clave para interpretar los listados

Todos los listados que se publican en esta Revista han sido preparados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los listados en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los programas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas que aparecen del teclado, se han sustituido los símbolos gráficos que aparecen normalmente en los listados por una serie de letras entre corchetes [ ] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla para aclarar la interpretación de las indicaciones entre corchetes:

[C8SHD] = Tecla cursor hacia abajo (con SHIFT)  
 [C8SHU] = Tecla cursor hacia arriba (con SHIFT)  
 [C8SHR] = Tecla cursor a la derecha (con SHIFT)  
 [C8SHL] = Tecla cursor a la izquierda (con SHIFT)

[HOME] = Tecla CLR/HOME (con SHIFT)  
 [CLR] = Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [VIS] corresponden a la pulsación de las teclas de 1 a 5 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOFF] respecto a la tecla CTRL y las teclas 9 y 10.

El resto de las indicaciones constan de la parte COMM + SHIFT seguida de una letra, número o símbolo —por ejemplo [COMM-] o [SHIFA]—. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE + + o SHIFT y A, respectivamente.

En los signos gráficos además se cuenta el número de veces que aparece. Por ejemplo, [7 CDRRR] equivale a 7 pulsaciones de la tecla cursor a la derecha y [3 SPC] tres pulsaciones de la barra espaciadora. \*

## INDICE DE ANUNCIANTES

	Pág.		Pág.
ALBAREDA.....	15	ITAR.....	33
ASTOC - DATA.....	47	LOBERCIO - INFO-IMPORT.....	52
BASIC MICROORDENADORES.....	79	MICROELECTRONICA Y CONTROL.....	44, 80
BOUTIQUE COMMODORE WORLD.....	27	MICROS GARDEN.....	47
CASA DE SOFTWARE.....	2, 7, 37, 46	MICRO WORLD.....	46
CCL.....	55	NUEVA ONDA.....	47
CENTRO DE INFORMATICA.....	46	OFERTA. Venta de muebles de oficina.....	43, 48
CIMEX ELECTRONICA.....	47	OFERTA. Venta de ordenador.....	54
COMMODORE WORLD (Distribución).....	58	PC WORLD.....	13
COMMODORE WORLD (Suscripciones).....	31	POKE.....	45
DATAMON/RITEMAN.....	4	RADIO WATT.....	46
ELECTROAFICION COMPUTER.....	46	REGISTER LATELY CONTINENTAL.....	49
ELECTRONICA SANDOVAL.....	46	REM.....	46
FERRE MORET.....	40, 41	SCS.....	17, 20, 21
FIRST.....	11	SEINFO.....	25
GENERALITAT DE CATALUNYA.....	34, 35	SOFTWARE ESPAÑA.....	46
HANTAREX.....	77	TELE SANT JUST.....	53
IEESA - MICROTERSA.....	23, 47	VENTAMATIC.....	46

# Contabilidad - 64 así de fácil.

El programa Contabilidad 64 es líder en ventas, por su sencillez, rapidez, eficacia y precio.

COMO INICIAR LA ENTRADA DE DATOS EN EL PLAN CONTABLE

Seleccione el número 8 de «menú general» y aparecerá en pantalla el submenú «utilidad». Llame al número 1 y haga una copia del disco. El original no se debe utilizar como disco de trabajo.



Comptabilidad S4 se ha realizado según el Plan General Contable Español y ha sido desarrollado especialmente para los intermediarios Commodity S4 y E&E.

\* El programa tiene capacidad para 300 usuarios y un número limitado de apuntes o registros por usuario; el programa permite generar nuevos discos en los que continuar el aprendizaje.

[illegible]

1. **Principios que componen la contabilidad y su estructura.**
2. **Cómo se estructura el programa Contabilidad III.**
3. **Cómo iniciar la entrada de datos en el Plan Contable.**
4. **Métodos para la introducción de partidas contables y obtención de resultados.**
5. **Balance de sumas y saldos y de situación.**
6. **Cómo consultar los cuantos y obtención del Mayor.**
7. **Cuentas de explotación y Cuentas de cierre.**

Seguidamente selección el número 4 «mantenimiento de ficheros» y fíjate la opción 1 «parámetros», aparecerá «mantenimiento de ficheros» con el cursor intermitente en el lugar donde se debe colocar el nombre de titular de la contabilidad. Mediante la tecla «return» desconectaré de línea en línea y automáticamente se irán actualizando los cambios.

[illegible]

Account	Debit	Credit	Balance
Jan 1, 2018			
Cash	100.00		100.00
Accounts Receivable	200.00		200.00
Inventory	300.00		300.00
Prepaid Insurance	100.00		100.00
Equipment	500.00		500.00
Accounts Payable		150.00	(150.00)
Owner's Equity		1,150.00	1,150.00
Total	1,200.00	1,200.00	

Para dar de alta las cuentas (tres dígitos) y subcuentas (seis dígitos) seleccione la opción 2 +cuentas-

Si prefieres conocer todas las ventajas hoy, envíame el cupón adjunto y recibiré gratuitamente el manual resumido de Contabilidad 44

casa de  
software s.a.

c/f. aragón, 372, 8.º, 6.º  
tel. 215 69 52  
95007 hancoluna

ORDER RECEIPT INFORMATION

Information on the use of the model can be found at <http://www.chem.mcgill.ca/~chem222/chem222.html>.

[illegible]

Copyright © 2004 John Wiley & Sons, Ltd.

**L**a flexibilidad y el bajo precio del C-64 no son las únicas cosas que le convierten en una máquina de hacer música de alta calidad. El secreto es un pequeño trozo de arena preparada llamado el SID (Sound Interface Device) Interface de Sonido.

El SID es uno de los más sofisticados sintetizadores de música disponibles en cualquier ordenador personal. Dispone de tres voces, cada una con su envolvente correspondiente de ataque/decaimiento, sostenimiento/relajación (ADSR); filtros, modulación de timbre y sincronización; y cuatro generadores de forma de onda. Pero esto no es todo.

Casi todos los ordenadores son capaces de producir algún tipo de sonido, pero, a diferencia de otros generadores, todas las características del SID pueden ser manipuladas, modificadas y controladas desde el teclado, sin la necesidad de usar más hardware. Como por arte de magia, el SID produce prácticamente cualquier cosa que tú puedas imaginar.

Para poder aprovechar al máximo ese sintetizador en un chip, tienes que saber algo acerca de la música y la programación en Basic. Las cosas se ponen más difíciles si no sabes nada de la música y menos todavía de la música. Sigue leyendo —aunque ritmo te suena a coche y bacle a pedalar— estamos aquí para ayudarte.

Aprender a tocar el sintetizador de la misma forma en que aprendes a tocar cualquier otro instrumento musical: prácticas.

Teclas NEW, pulsa la tecla Return y teclas el comando L. Sálvalo en disco o cinta antes de ejecutarlo.

Se utiliza la sentencia Poke con frecuencia. Antes de seguir leyendo este artículo, te aconsejo que leas el apartado sobre Pokes y Pokes.

#### Descripción del Programa

El programa empieza en la línea 30 con



un REM. En la línea 20, se establece que S representa 54272 durante el resto del programa. A continuación se apaga el chip SID introduciendo ceros mediante unos Pokes en todos los registros que controlan un bacle For...Next. Durante la primera pasada del bacle, I es igual a 0 (FOR I=0 TO 255); se añade a S (S=S+I), y se introduce mediante un Poke en 54272 (POKE S+I,0). Cuando el programa encuentra la sentencia Next, vuelve automáticamente a la última sentencia For y aumenta el valor de I por 1, la continuación 0 se introduce en 54273 (S+1) mediante un Poke. Cuando se vuelve a encontrar la sentencia Next, I se convierte en 2, un cero se introduce en 54274 mediante un Poke, etc., hasta que el chip entero quede borrado o apagado.

La línea 30 usa un Poke para introducir la envolvente ataque/decaimiento y sostenimiento/relajación en los registros correctos para la voz 1. Puedes modificar el tipo de sonido cambiando los números que siguen S+5 o S+6. La A/D y S/R, utilizada aquí junto con la onda de diente de sierra, producen un sonido parecido a un bacle.

En la línea 40 el volumen interno sube hasta 15, su punto máximo. Ajusta el volumen en tu televisor o monitor hasta encontrar un nivel cómodo.

#### Sentencias Read y Data

Cuando el C-64 encuentra una sentencia Read (línea 50), busca automáticamente una sentencia Data que le acompañe (que puede estar en cualquier lugar del programa, en este caso, las líneas 200-300). Las sentencias Read y Data trabajan en equipo: una no puede existir sin la otra. Observa que la sentencia Read va seguida de dos nombres de variables: HF, que contiene 16, y LF, que contiene 195. Imagínate que son etiquetas en las cajas que contienen los datos.

Cuando se encuentra la sentencia Read, seguida de dos variables, el ordenador mira la primera sentencia DATA que encuentre en el programa y recoge los dos primeros caracteres después de la palabra DATA. El ordenador sabe que los datos van separados por comas.

De momento saltar la línea 60. La línea 70 usa un Poke para introducir los datos contenidos en HF y LF en los registros de alta y baja frecuencia para la voz 1. Observa que READ HF se encuentra primero (línea 50), pero el ordenador va primero a POKE LF (línea 70). Yo podía haber invertido el orden de esto, pero quiero que veas que algo parece ocurrir al revés en un ordenador. Lo importante es usar un poke para introducir el valor de HF en el registro de alta frecuencia y el valor de LF en el registro de baja frecuencia. Si se hace al revés, se producen chidos en vez de tonos cuando ejecutas el programa.

¿Qué ha pasado hasta ahora? Se ha borrado el chip SID, fijado la envolvente

# SONIDO

*Casi todos los ordenadores son capaces de producir algún tipo de sonido, pero, a diferencia de los otros, el C-64 dispone de un sonido que puede ser manipulado, cambiado y controlado desde el teclado —no necesita más hardware—.*

ADSR, encendiendo el volumen e introduciendo los números que controlan la nota. Pero no suena nada todavía.

El sonido se produce a partir de la línea 80. En esta línea se selecciona y se enciende la forma de onda, y hasta que no se haga esto, no se produce ningún sonido.

Cuando se ejecuta la línea 80, suena una nota. Un bacle de retraso en la línea 90 hace que la nota no suene para siempre. Le indica al ordenador que cuente 200 "jiffies" antes de ejecutar la siguiente instrucción. Un "jiffie" es un 1/60 de segundo en tiempo real, de modo que 200 "jiffies" no es mucho tiempo. Después de contar hasta 200, el programa va a la línea 100, donde el sonido se apaga introduciendo 32 en S+4 mediante un Poke, lo que quita el 1 que enciende el sonido en la línea 80. Se sigue generando la onda de diente de sierra, pero ahora no se oye. Después de apagar el sonido, el ordenador cuenta 10 "jiffies" (FOR I=1 TO 10 NEXT I) antes de pasar a la siguiente instrucción. Esto intervala un silencio muy breve entre los tonos.

#### El Bacle GOTO

El principio de un bacle GOTO se encuentra en la línea 110. Mientras que los números de línea establecen el orden en que el ordenador ejecuta las instrucciones, un GOTO interrumpe dicho orden y envía al programa a una línea que queda fuera de secuencia. GOTO 50 significa volver a la línea 50 y ejecutar las instrucciones que se encuentran allí. De esta forma, se vuelve a iniciar el proceso Read/Data, pero esta vez con la segunda unidad de datos, ya que esta es la segunda vez que se ha ejecutado la instrucción Read en la línea 50. El programa sigue

C-64  
VIC-20

# MUSICA

ejecutando este bucle GOTO hasta que termine con todos los datos, pero daría un Syntax Error si no estuviera la línea 60 (la que saltaste antes).

## El poder de HF...Then

Un ordenador programado en Basic puede tomar decisiones gracias a la sentencia IF...Then. Funciona de la siguiente forma: Si (IF) algo es verdad, entonces (Then) hay que hacer una cosa determinada. Cuando la condición comprobada no es correcta cuando HF no es igual a -1 en este programa), se ignora la instrucción que sigue THEN (junto con el resto de la línea). La línea 60 de este programa comprueba si el valor de HF es igual a -1, que se utiliza como una bandera. Ahora acordate, cada vez que se leen nuevos datos, y el programa vuelve a la línea 50, se modificará el valor en las cajas etiquetadas HF y LF. Y cada vez, el valor de HF será comparado con -1 gracias a la línea 60. Si lees la línea 200, verás que los dos últimos datos son -1, -1. Se necesitan dos unos (1s) negativos porque el ordenador recoge dos grupos de datos a la vez (READ HF, LF), y el programa dejara de ejecutarse y te enviará un mensaje de error si no puede recoger los dos. Cuando lees este último par de datos, volverá a comprobar si HF es igual a -1, y esta vez será cierto. A continuación, y siguiendo la instrucción después de THEN en la línea 60, se apagará el volumen y se finalizará el programa. Esta es la forma de salir de un bucle GOTO.

## Jugando un poco

Cuando tienes el programa funcionando, excóballo y experimenta un poco, produciendo diferentes sonidos. No puedes dañar el ordenador introduciendo datos a través del teclado, y si el ordenador queda colgado (desaparece el cursor), apágalo durante unos segundos. A continuación se presentan algunas técnicas para modificar el programa:

1) Modificar los datos para que suenen tonos diferentes.

Mira "La Guía del Usuario de Commodore" (la que recibiste junto con el ordenador) y consulta al Apéndice M.

Valores de las Notas Musicales, en la página 152. Tal y como está el programa ahora, suena una octava más alto, empezando con C-4 (el do de la escala central). Los datos para el programa se encuentran en la página 153. Busca C-4 en la columna Nota-Octava y sigue hacia la derecha en las columnas de Alta Frecuencia y Baja Frecuencia, donde encontrarás los dos primeros datos de la línea 200. Puedes cambiar los datos (y hasta añadir más líneas al programa) con tal de que tengas cuidado de mantener las parejas

de HF y LF y que los dos últimos datos sean -1.

2) Cambiar la forma de onda

Este programa utiliza una onda de diente de sierra (valor 32), que se activa en la línea 80 y se desactiva en la línea 100. Si cambias la forma de onda, se cambiará la calidad del sonido. Prueba con la forma de onda triangular (valor de 17 en la línea 80 y de 16 en la línea



Por Perry HERRINGTON  
Adaptado por Abner J. B. B. E.  
Traducido por Valerie STANKS

Commodore World 1166 1987/8

100) o el ruido (129 en la línea 80, 128 en la línea 100). Puedes usar la onda de pulso también (valor de 65 en la línea 80 y de 64 en la línea 100), pero añade esta línea al programa:

35 POKE S+3,10:POKE S+4,150:REM ANCHURA DE PULSO

1) Cambiar el generador de envolvente ADSR.

Se pueden imitar diferentes instrumentos musicales si tienes los valores ADSR o simplemente experimentando con sonidos diferentes. POKE S+5 y el número de AD y S+h con el número de SR (arriba en la línea 30), utilizan la onda de diente de sierra (valor 32) para conseguir los sonidos de los siguientes instrumentos: bajo: AD=3,SR=0 (tal y como está en el programa original), piano AD=0,SR=9, órgano AD=0,SR=240. Utiliza la onda rectangular con la línea 15 y AD=9,SR=0 para la guitarra. Prueba con valores ADSR con otras formas de onda para conseguir más sonidos distintos.

*Se habla mucho de la capacidad musical del C-64, pero el VIC-20 también es capaz de tocar un par de cosas.*

*A continuación veremos cómo sacarle la música al VIC-20*

**H**ace un par de años, mi marido trajo a casa un aparato apertamente inocente.

"¿Que te has comprado un qué?" le dije. Lo estaba encubriendo a la televisión mientras yo me alejaba a toda velocidad.

"Yo creo que te va a gustar", me dijo. "Hace música".

No me gustó. Me encantó, incluso cuando todo el mundo empezó a volverse loco con el sintetizador del C-64,

yo me mantuve firme en defensa de mi pequeño VIC.

Para que el VIC-20 produzca sonidos, tienes que saber latido de música como de programación Basic.

El VIC dispone de cuatro voces —tres de ellas son musicales y una es ruido blanco (lo que se oye entre canal y canal en una televisión). No hay muchos controles de sonido complicados, y dispones de todo lo necesario para hacer música y efectos de sonido. Existen cinco posiciones de memoria (también llamadas registros o bytes) dedicadas al sonido (ver la Tabla 2).

Los tres primeros registros en la Tabla 2 controlan las notas musicales, los tonos que componen la melodía y la armonía. A propósito, C4 es el do de la escala central. Cada voz tiene un rango de tres octavas (una octava consiste en 12 tonos consecutivos), pero dado que se solapan, el rango completo del VIC —es decir, desde el tono más bajo hasta el más alto— son cinco octavas. Si tienes un piano, ignora la octava más baja y las dos más altas, y lo que queda es el rango del VIC.

Para que el VIC produzca sonido, hay que activar el volumen interno del ordenador. Uno se hace introduciendo un número de 1 a 15 mediante un poke en el registro 36878. (Antes de continuar con este artículo, lee la sección dedicada a pokes y peeks). Después de activar el volumen, el número correspondiente a la nota elegida se introduce en uno de los registros de voz mediante un poke. Para que toque el do de la escala central (C4) en la voz baja, teclea NLW, pulsa la tecla Return y teclea las siguientes líneas:

POKE 36878,15  
POKE 36874,240

La forma más rápida de pasar el sonido es pulsando las teclas Run/Stop y Restore a la vez. Esto hace un Reset en los puertos del VIC y borra la pantalla. Otro método es usar las teclas del cursor y sustituir 240 por 0, pulsando la tecla Return mientras el cursor se encuentra todavía en esa línea. También puedes apagar el volumen (POKE 36878,0) o usar un Poke para introducir un número menor que 128 o mayor que 255 en el registro de voz. (Si no se produce ningún sonido cuando tecleas las líneas, comprueba el volumen de tu televisor o monitor).

Ahora escucha el do de la escala central en la segunda voz:

POKE 36878,15  
POKE 36875,225

Ahora prueba con el do de la escala central en la tercera voz:

POKE 36878,15  
POKE 36876,195

Si escuchas con atención, te darás cuenta de que estos tonos suenan un poco diferentes en cada voz. La nota es la misma (es el do de la escala central),

aunque los números 240, 225 y 195 son distintos (porque los rangos de las voces se solapan), pero la calidad del sonido, lo que los músicos llaman color del tono o timbre, no es exactamente la misma. Las voces 1 y 2 son más suaves que la tercera. Usa para que la melodía, que normalmente se toca en tonos más altos cuando se sincroniza con otras voces, se destaque sobre ellas.

El VIC-20 es capaz de producir muchas notas que un instrumento acústico no puede. Hasta ahora el VIC se ha estado cantando en modo directo. Ahora vamos a cambiar al modo programa para escuchar todas las notas (el rango) que tiene el registro más bajo del VIC. Borra la memoria tecleando NEW y pulsando Return, y teclea el siguiente programa. Teclea incluso los REM (el ordenador ignora todo lo que vaya detrás de un REM en una línea) hasta que comprendas lo que está ocurriendo en el siguiente programa:

```
10 S=36874 V=36878 T=128
20 POKE V:REM ENCENDER VOLUMEN
30 POKE S:T:REM SONIDO
40 FOR D=1 TO 100 NEXT D:REM BUCLE DE DURACION DEL SONIDO
45 POKE S:FOR T=1 TO 50 NEXT T:REM BUCLE DE DURACION DEL SILENCIO
50 T=T+1:REM INCREMENTAR NOTA
60 IF T>253 THEN POKE S:END:REM COMPROBAR FIN
70 GOTO 30
```

Salva el programa antes de ejecutarlo. La primera línea inicializa las variables y las constantes. Podías haber elegido cualquier letra (o combinación de dos letras o una letra y un número) para representar la voz (S), volumen (V) o el tono (T), pero esas letras se ayudarán a recordar lo que va en el programa más tarde. El uso de las variables, que normalmente cambian de valor a lo largo de un programa, y las constantes, que no se cambian, en vez de números de registro, tiene varias ventajas. Son más fáciles de leer, lo que reduce los riesgos de cometer un error, utilizan menos memoria y el VIC trabaja más rápido con ellas que con números.

#### Bucle For...Next

Se enciende el volumen en la línea 20. La línea 30 produce el primer sonido audible mediante un poke (el ordenador sustituye automáticamente el valor numérico actual por una variable en un programa, en este caso 128) en el registro de la voz 1 (36874). El bucle For...Next en la línea 40 le indica al ordenador que cuente hasta 100 "jiffies" antes de ejecutar la siguiente instrucción. De esta forma se controla el tiempo que dura el sonido (quita la línea 40 para ver qué ocurre).







## **E**l VIC dispone de cuatro voces —tres de ellas

*son musicales y una es ruido blanco (lo que se oye entre canal*

*y canal en una televisión). No hay muchos controles*

*de sonido complicados, y dispone de todo lo necesario*

*para hacer música y efectos de sonido.*

► Funciona de la siguiente forma: el VIC reconoce la sentencia For y la variable de control que la sigue (D), que en este caso significa "delay" (retraso). D asume el valor del primer número después del signo igual a  $D=1$  y el ordenador busca la siguiente instrucción que, en este caso, es NEXT D. Cuando reconoce esto, el VIC-20 vuelve automáticamente a la sentencia For y la variable D se incrementa por 1. Continúa buscando la siguiente instrucción —NEXT D— hasta que D es igual al número que sigue TO en la sentencia. Este proceso se llama bucle. Cuando D finalmente queda igual a 100, el ordenador pasa a la siguiente instrucción después de NEXT D y continúa ejecutando el programa.

Este bucle de retraso es un ejemplo de la aplicación más sencilla de la sentencia For...Next en Basic. Se puede llegar a hacer una programación más complicada con un bucle For...Next intercalando instrucciones entre FOR y NEXT y coordinando la variable de control (D) y el rango (1 a 100) con otros aspectos de un programa. Los bucles For...Next también pueden ir anidados, uno dentro de otro. Si a todo esto se suma el hecho de que los bucles

For...Next se usan con frecuencia —hay otro en la línea 45 de este programa— y podrás ver que es una herramienta muy útil para la programación en Basic.

Sin embargo, antes de que se encuentre el segundo bucle For...Next en la línea 45, se le indica al ordenador que apague el sonido (POKE 5,0). Después de hacer esto, cuenta hasta 50, ahora usando la variable I, para que se produzca un breve silencio entre las notas. Un segundo consiste en 60 "jilfis" (el VIC-20 dispone de un reloj interno) y podrías modificar el número de "jilfis" en los bucles de retraso de este programa, para hacerle una idea de este modo de cronometrar.

Acuédate, T representa Tono y fue inicializado a un valor de 128 en la línea 30. La sentencia  $T=T+1$  NO es una cuestión algebraica. Es una vieja técnica de programación que antiguamente iba precedida por la palabra LET (Dejar). Si tienes esto en cuenta, entenderás que la sentencia significa "DEJA que el valor de T tenga el valor actual más uno". Esto significa que T tiene un valor de 129 después de la línea 50.

### **Bucles GOTO**

De momento saltate la línea 60. El orden en que las instrucciones se ejecutan en Basic está controlado por la numeración de la línea del programa. A veces es necesario que las instrucciones se repitan o que se ejecuten en un orden diferente del que queda establecido por los números de línea. La instrucción de la línea 70 es un ejemplo de otro tipo de bucle, un bucle GOTO, que realiza ambas cosas. El VIC-20 reconoce que el GOTO en la línea 70 significa "vete a la línea 30 y sigue las instrucciones que encuentres allí". ¿Se le puede decir a un ordenador que vaya a cualquier número de líneas de un programa, tanto hacia adelante como hacia atrás? Así que el VIC hace el bucle para volver a la línea 30 y suena otra nota. Cada vez que esto ocurre, se le suma uno a T, de modo que el tono de la nota incrementa un poco. El bucle GOTO incluye todo lo contenido entre las líneas 30-70.

La creación de un bucle GOTO es complicada y te puede traer problemas si no cuidas mucho su estructuración. Por ejemplo, sin la línea 60, el bucle

creado en la línea 70 sería infinito. Solo se podría parar interrumpiendo el programa manualmente. En este programa, hay dos cosas que impiden que esto ocurra, y una de ellas no es una instrucción.

Cuando el bucle GOTO se ha ejecutado 128 veces, T se habrá incrementado hasta tener un valor de 256, y cuando el ordenador intenta introducir 256 mediante un poke en el registro de sonido (o cualquier otro registro), se parará la operación y enviará un mensaje de error. Ningún registro de su procesador de 8 bits (como el VIC-20) puede funcionar con un número mayor que 255. Pero como no va a ocurrir ahora gracias a la instrucción de la línea 60.

### **II... Then de nuevo**

Esta sentencia Basic tiene el poder de tomar una decisión. El ordenador comprobará la condición presentada por la sentencia If, y solamente si encuentra que es verdad ejecutará la instrucción que sigue la sentencia Then. Por lo tanto, cada vez que se pasa por el bucle GOTO, el ordenador comprueba para ver si T es mayor que 253. Si no lo es, el ordenador ignora el resto de la línea.

(Posiblemente hayas notado un error en los números. Dije que el máximo valor permitido para un Poke era 255 —y lo es— pero el programa comprueba para ver si T es igual a 253. Intenta sustituir el valor de 253 por 255 para ver lo que ocurre). Cuando T finalmente sea mayor que 253 —cuando  $T=254$ — se ejecuta la instrucción que sigue a THEN. El bucle GOTO, que podía seguir ejecutándose indefinidamente, se da por finalizado, y termina el programa.

Intenta modificar partes del programa para producir sonidos diferentes. Prueba borrando la línea 45 (utiliza las teclas del cursor para introducir los REM detrás de los números de línea, y no tendrás que volver a teclear la línea para volver a activarla; simplemente quita la sentencia REM). Al omitir esto, se oirá un barrido del generador de notas en vez de notas en vez de notas individuales.

Incrementa T en la línea 50 por un número mayor para dejar más espacio entre cada nota; por ejemplo, puedes poner  $T=T+2$ . Quita la línea 50 para crear un bucle indefinido en el que se repite la misma nota una y otra vez. (Se da el mismo resultado cambiando la línea 70 a GOTO III. ¿Sabes por qué?). Además, puedes trasladar el valor del altavoz (S) a otro de los registros de sonido (S=36875, 36876 o 36877) y tomar nota de las diferencias.

### **Los Pokes y Peks de Commodore**

Tienes que usar los Pokes en el C-64 para que cante el SID. Cuando se utiliza ►



# PC WORLD/ESPAÑA

## es tu pasaporte al "Mundo" del PC



PRECIO ESPECIAL  
COMODORE WORLD VIGENTE  
HASTA EL 15-MAYO-1984

**Te presentamos un nuevo "WORLD"**

*Oferta especial para los suscriptores  
de "Comodore World"*

*Si eres lector de "Comodore World" y te interesa el mundo de los PCs, envíanos el boleto adjunto y un talón (o giro) por 250 pesetas y recibirás el primer número de "PC World" a mitad de precio (P.V.P. 500 pesetas).*

Enviar a: PC WORLD • C/ Barquillo, 21 - 3ª Izda. • 28004 Madrid

DESEARIA ME ENVIASEN LA REVISTA PC WORLD

NOMBRE .....  
EMPRESA .....  
DIRECCION .....  
POBLACION .....  
(C.P.) ..... PROVINCIA ..... TEL.F. ....

Desearia me envíen ..... ejemplares a 250 Ptas. por ejemplar

Suscripción de 6 números por 1.500 Ptas.

Envío talón a: ..... o giro a: ..... por ..... Ptas.

Desearia recibir tarifas de publicidad .....

En Pakistán, un ordenador Commodore, votó durante una ordenación electrónica, permitiendo a una industria de algodón ser reconocida al estar por la lista Ricam. Tanto el método de elección, como el sistema de votación, son programas. El ordenador emitió el formato Pak, (posicionamiento, justificación, validación, verificación) correspondiente a los cuatro tipos de ordenes del RVM. Memorizada. A través de la interfaz de datos se transmitieron las instrucciones por medio de las líneas de RVM al Memorizador de Señal de Datos, el cual por su orden de trabajo, según el sistema de control, envió a los tres formatos de salida.

[illegible]

A partir de este análisis se establecieron 5 grupos de riesgo de SIDA en la población de estudio, de acuerdo a los resultados de las pruebas de laboratorio y de la historia clínica. Los grupos de riesgo se definieron de la siguiente manera:

- Grupo 1: personas que no han tenido relaciones sexuales con un compañero sexual masculino.
- Grupo 2: personas que han tenido relaciones sexuales con un compañero sexual masculino, pero no han usado condón.
- Grupo 3: personas que han tenido relaciones sexuales con un compañero sexual masculino, pero no han usado condón y han tenido relaciones sexuales con más de un compañero sexual masculino.
- Grupo 4: personas que han tenido relaciones sexuales con un compañero sexual masculino, pero no han usado condón y han tenido relaciones sexuales con más de un compañero sexual masculino y han tenido relaciones sexuales con un compañero sexual masculino que pertenece a uno de los grupos de riesgo.
- Grupo 5: personas que han tenido relaciones sexuales con un compañero sexual masculino, pero no han usado condón y han tenido relaciones sexuales con más de un compañero sexual masculino y han tenido relaciones sexuales con un compañero sexual masculino que pertenece a uno de los grupos de riesgo y han tenido relaciones sexuales con un compañero sexual masculino que pertenece a uno de los grupos de riesgo.

Cuando haces un Peek en una pestaña de la interfaz, estas accedendo solo a un fragmento de la Peek. Te estás preguntando si estás viendo lo que deseas. El valor almacenado es un signo de determinación. No se puede hacer un Peek en los registros de la BMD de 54272-54296 (de 54296-54320) de ningún registro de solo escritura. Se puede hacer un Peek en los registros de solo lectura PRIN1-PRIN6, 54297-54320 y PRIN1-PRIN6, 54321-54384. Sin embargo, esto no garantiza que el contenido de memoria que se puede hacer un Peek. La característica de que el BMD es de solo lectura con respecto a la memoria de las pestañas de memoria se puede hacer un Peek en el BMD.

In this example (PUK1 8-4.33), we start the initial value,  $y(0)$ , on PUK1

de un valor de 33 en la posición de memoria 54276. Vamos a ver lo que significa este 33.

## Data, Dates, &amp; Symbols

Un byte es lo que hemos estado llamando una posición de memoria, o un registro (542<sup>2</sup> es el primer byte del chip de SMD). Pero la unidad más pequeña dentro de todos los ordenadores no es un byte, sino un bit, y hay ocho bits en cada byte (el duple bit = medio byte, es, para un modelo).

Mirando la tabla de la carpeta SII0 se observa la poca población más pequeña de la detención, representa un 60,30 por ciento de la muestra total, el resto hacen a su izquierda. Los dos primeros en la lista son el condenado argentino los votos según la detención de memoria por ejemplo, 542, los 8-14. Aves si pueden encontrar en los libros, en la tabla. El primer por la izquierda en los libros, como de Minnesota del SII0 los, los, hacia abajo en la columna para la voz 1. Los 14 que, según la 8-14. Mirando hacia la columna del Nombre de Rigodon, columna, se llama el registro de centros. Significa la detención, pueden ver los libros, que comienzan el libro, en 542-76.

Observando las voces 2, 3 y 4 en las columnas a, b, c, d, e y f, el registro 1 es, como afirmamos, una voz  $\bar{a}$  que afirma todas las voces pares. La parte superior de esta tabla representa algunas registros que afirman la voz 1, 2 o 3. Note registro afirmando cada voz de forma independiente de modo que se afirme siete a cada diez y cinco para la voz 1 (es-4-5), 8-9-1, para seguir al registro de control para la voz 2. Anade otros seis para la voz 3.

La fila superior de posiciones contiene los números utilizados para actualizar otros determinandos. En la fila que se está desarrollando,  $i$  es uno de los valores determinandos antes, ya mencionados, que se actualizan mediante un PoE en las posiciones de memoria. Con esta información, puedes interpretar el valor de  $U$  en la instrucción  $INSTR(i, k, 2, 1)$ .

En este caso, 34 es la suma de la columna 52 y la columna 1. Se está diciendo al C64 que quieres que utilice la forma de onda de diente de sierra (un valor de 127) que active el sonido (con un valor de 0 para la no utilización de memoria 54276). POKé 8+4,55, es decir indirecto al ordenador. Pateo de la línea la voz (se analiza un valor de 0 cuando la instrucción POKé 8+4,32. Aunque se haya apagado el sonido, se sigue produciendo la forma de onda de diente de sierra, lo que pasa es que tu no lo oyes.

**Activado.** Nos va a explicar ahora todos los detalles del sistema binario. Te va a decir cómo el sistema binario, el que utilizan los ordenadores, se basa en las potencias de 2. La Tabla 3 demuestra los números de bits de un byte que aumentan de derecha a izquierda, junto con las correspondientes potencias de 2. A su vez, a la izquierda del sistema de

Ten en cuenta que los números de los bits son las potencias correspondientes de 2. Calcula 2 elevado a la potencia de 5 ( $2^5 = 2 \cdot 2 \cdot 2 \cdot 2 \cdot 2$ ) y da 32, una parte del número decimal utilizado para hacer un Pake en la posición de memoria de nuestro ejemplo. Sección 6.1 de la memoria 5 introducción al número 32 mediante un Pake en el registro 14 exponenciación en el 0-64 se realiza con la tecla de flecha arriba. Para conseguir el resultado de 2 elevado a la potencia de 5 en modo directo, pulsa PRINS 2 (flecha arriba) 5 y pulsa la tecla Return.

El número más alto que se puede introducir en un bit medaalti un Poké es  $255 \cdot 128 + 64 \cdot 12 + 16 \cdot 4 + 2$ , es igual a 255, esto activa todos los bits de este byte. Los números menores que 255 tienen que ser divididos en dos bytes. Para saber los bits que son activados por el valor de un Poké, resta la máxima potencia de 2 (en forma decimal) que es igual a o menor que el mismo número del Poké. Llamamos que el valor del Poké es 133, 133 menos 128 (que activa el bit número 7) deja 5. 5 menos 4 (que activa el bit número 2) deja 1, que activa el bit número 0. Por lo tanto el Poké 133 activa los bits 7, 2 y 0. El 133 como número binario es 100001101.

A partir de ahora, cuando veas la instrucción "para producir una onda de choque de sexta vez" tienes que activar los buses 0 y 5 en el Registro de Control del S60 para la Voz 1. Consulta las Tablas 3.1 y 3.2 convertiendo en PPSK:  $S = 4.5^\circ$ . Se practican así poco a los mejor terminas sabiendo lo que estás haciendo.

## VIII 201 A.C.-64

```

100  def __init__(self, num):
101      self.num = num
102      self.__dict__['_num'] = num
103      self.__dict__['_num2'] = num * 2
104      self.__dict__['_num3'] = num * 3
105      self.__dict__['_num4'] = num * 4
106      self.__dict__['_num5'] = num * 5
107      self.__dict__['_num6'] = num * 6
108      self.__dict__['_num7'] = num * 7
109      self.__dict__['_num8'] = num * 8
110      self.__dict__['_num9'] = num * 9
111      self.__dict__['_num10'] = num * 10
112      self.__dict__['_num11'] = num * 11
113      self.__dict__['_num12'] = num * 12
114      self.__dict__['_num13'] = num * 13
115      self.__dict__['_num14'] = num * 14
116      self.__dict__['_num15'] = num * 15
117      self.__dict__['_num16'] = num * 16
118      self.__dict__['_num17'] = num * 17
119      self.__dict__['_num18'] = num * 18
120      self.__dict__['_num19'] = num * 19
121      self.__dict__['_num20'] = num * 20
122      self.__dict__['_num21'] = num * 21
123      self.__dict__['_num22'] = num * 22
124      self.__dict__['_num23'] = num * 23
125      self.__dict__['_num24'] = num * 24
126      self.__dict__['_num25'] = num * 25
127      self.__dict__['_num26'] = num * 26
128      self.__dict__['_num27'] = num * 27
129      self.__dict__['_num28'] = num * 28
130      self.__dict__['_num29'] = num * 29
131      self.__dict__['_num30'] = num * 30
132      self.__dict__['_num31'] = num * 31
133      self.__dict__['_num32'] = num * 32
134      self.__dict__['_num33'] = num * 33
135      self.__dict__['_num34'] = num * 34
136      self.__dict__['_num35'] = num * 35
137      self.__dict__['_num36'] = num * 36
138      self.__dict__['_num37'] = num * 37
139      self.__dict__['_num38'] = num * 38
140      self.__dict__['_num39'] = num * 39
141      self.__dict__['_num40'] = num * 40
142      self.__dict__['_num41'] = num * 41
143      self.__dict__['_num42'] = num * 42
144      self.__dict__['_num43'] = num * 43
145      self.__dict__['_num44'] = num * 44
146      self.__dict__['_num45'] = num * 45
147      self.__dict__['_num46'] = num * 46
148      self.__dict__['_num47'] = num * 47
149      self.__dict__['_num48'] = num * 48
150      self.__dict__['_num49'] = num * 49
151      self.__dict__['_num50'] = num * 50
152      self.__dict__['_num51'] = num * 51
153      self.__dict__['_num52'] = num * 52
154      self.__dict__['_num53'] = num * 53
155      self.__dict__['_num54'] = num * 54
156      self.__dict__['_num55'] = num * 55
157      self.__dict__['_num56'] = num * 56
158      self.__dict__['_num57'] = num * 57
159      self.__dict__['_num58'] = num * 58
160      self.__dict__['_num59'] = num * 59
161      self.__dict__['_num60'] = num * 60
162      self.__dict__['_num61'] = num * 61
163      self.__dict__['_num62'] = num * 62
164      self.__dict__['_num63'] = num * 63
165      self.__dict__['_num64'] = num * 64
166      self.__dict__['_num65'] = num * 65
167      self.__dict__['_num66'] = num * 66
168      self.__dict__['_num67'] = num * 67
169      self.__dict__['_num68'] = num * 68
170      self.__dict__['_num69'] = num * 69
171      self.__dict__['_num70'] = num * 70
172      self.__dict__['_num71'] = num * 71
173      self.__dict__['_num72'] = num * 72
174      self.__dict__['_num73'] = num * 73
175      self.__dict__['_num74'] = num * 74
176      self.__dict__['_num75'] = num * 75
177      self.__dict__['_num76'] = num * 76
178      self.__dict__['_num77'] = num * 77
179      self.__dict__['_num78'] = num * 78
180      self.__dict__['_num79'] = num * 79
181      self.__dict__['_num80'] = num * 80
182      self.__dict__['_num81'] = num * 81
183      self.__dict__['_num82'] = num * 82
184      self.__dict__['_num83'] = num * 83
185      self.__dict__['_num84'] = num * 84
186      self.__dict__['_num85'] = num * 85
187      self.__dict__['_num86'] = num * 86
188      self.__dict__['_num87'] = num * 87
189      self.__dict__['_num88'] = num * 88
190      self.__dict__['_num89'] = num * 89
191      self.__dict__['_num90'] = num * 90
192      self.__dict__['_num91'] = num * 91
193      self.__dict__['_num92'] = num * 92
194      self.__dict__['_num93'] = num * 93
195      self.__dict__['_num94'] = num * 94
196      self.__dict__['_num95'] = num * 95
197      self.__dict__['_num96'] = num * 96
198      self.__dict__['_num97'] = num * 97
199      self.__dict__['_num98'] = num * 98
200      self.__dict__['_num99'] = num * 99
201      self.__dict__['_num100'] = num * 100
202      self.__dict__['_num101'] = num * 101
203      self.__dict__['_num102'] = num * 102
204      self.__dict__['_num103'] = num * 103
205      self.__dict__['_num104'] = num * 104
206      self.__dict__['_num105'] = num * 105
207      self.__dict__['_num106'] = num * 106
208      self.__dict__['_num107'] = num * 107
209      self.__dict__['_num108'] = num * 108
210      self.__dict__['_num109'] = num * 109
211      self.__dict__['_num110'] = num * 110
212      self.__dict__['_num111'] = num * 111
213      self.__dict__['_num112'] = num * 112
214      self.__dict__['_num113'] = num * 113
215      self.__dict__['_num114'] = num * 114
216      self.__dict__['_num115'] = num * 115
217      self.__dict__['_num116'] = num * 116
218      self.__dict__['_num117'] = num * 117
219      self.__dict__['_num118'] = num * 118
220      self.__dict__['_num119'] = num * 119
221      self.__dict__['_num120'] = num * 120
222      self.__dict__['_num121'] = num * 121
223      self.__dict__['_num122'] = num * 122
224      self.__dict__['_num123'] = num * 123
225      self.__dict__['_num124'] = num * 124
226      self.__dict__['_num125'] = num * 125
227      self.__dict__['_num126'] = num * 126
228      self.__dict__['_num127'] = num * 127
229      self.__dict__['_num128'] = num * 128
230      self.__dict__['_num129'] = num * 129
231      self.__dict__['_num130'] = num * 130
232      self.__dict__['_num131'] = num * 131
233      self.__dict__['_num132'] = num * 132
234      self.__dict__['_num133'] = num * 133
235      self.__dict__['_num134'] = num * 134
236      self.__dict__['_num135'] = num * 135
237      self.__dict__['_num136'] = num * 136
238      self.__dict__['_num137'] = num * 137
239      self.__dict__['_num138'] = num * 138
240      self.__dict__['_num139'] = num * 139
241      self.__dict__['_num140'] = num * 140
242      self.__dict__['_num141'] = num * 141
243      self.__dict__['_num142'] = num * 142
244      self.__dict__['_num143'] = num * 143
245      self.__dict__['_num144'] = num * 144
246      self.__dict__['_num145'] = num * 145
247      self.__dict__['_num146'] = num * 146
248      self.__dict__['_num147'] = num * 147
249      self.__dict__['_num148'] = num * 148
250      self.__dict__['_num149'] = num * 149
251      self.__dict__['_num150'] = num * 150
252      self.__dict__['_num151'] = num * 151
253      self.__dict__['_num152'] = num * 152
254      self.__dict__['_num153'] = num * 153
255      self.__dict__['_num154'] = num * 154
256      self.__dict__['_num155'] = num * 155
257      self.__dict__['_num156'] = num * 156
258      self.__dict__['_num157'] = num * 157
259      self.__dict__['_num158'] = num * 158
260      self.__dict__['_num159'] = num * 159
261      self.__dict__['_num160'] = num * 160
262      self.__dict__['_num161'] = num * 161
263      self.__dict__['_num162'] = num * 162
264      self.__dict__['_num163'] = num * 163
265      self.__dict__['_num164'] = num * 164
266      self.__dict__['_num165'] = num * 165
267      self.__dict__['_num166'] = num * 166
268      self.__dict__['_num167'] = num * 167
269      self.__dict__['_num168'] = num * 168
270      self.__dict__['_num169'] = num * 169
271      self.__dict__['_num170'] = num * 170
272      self.__dict__['_num171'] = num * 171
273      self.__dict__['_num172'] = num * 172
274      self.__dict__['_num173'] = num * 173
275      self.__dict__['_num174'] = num * 174
276      self.__dict__['_num175'] = num * 175
277      self.__dict__['_num176'] = num * 176
278      self.__dict__['_num177'] = num * 177
279      self.__dict__['_num178'] = num * 178
280      self.__dict__['_num179'] = num * 179
281      self.__dict__['_num180'] = num * 180
282      self.__dict__['_num181'] = num * 181
283      self.__dict__['_num182'] = num * 182
284      self.__dict__['_num183'] = num * 183
285      self.__dict__['_num184'] = num * 184
286      self.__dict__['_num185'] = num * 185
287      self.__dict__['_num186'] = num * 186
288      self.__dict__['_num187'] = num * 187
289      self.__dict__['_num188'] = num * 188
290      self.__dict__['_num189'] = num * 189
291      self.__dict__['_num190'] = num * 190
292      self.__dict__['_num191'] = num * 191
293      self.__dict__['_num192'] = num * 192
294      self.__dict__['_num193'] = num * 193
295      self.__dict__['_num194'] = num * 194
296      self.__dict__['_num195'] = num * 195
2
```

### Sikumbang, Ningsiliran

Según el sistema binario, el número 33 sería representado como 00100001, donde el 1 significa activado y el 0 des-



# MUSIC 64

## PERSONAL COMPUTER MUSIC



### Teclado "MUSIC 64"

El teclado "MUSIC 64" se conecta físicamente al COMMODORE 64, y consta de:

- 18 teclados de 4 octavas de Do a Do
- 16 interfaz
- 11 software necesario
- Se puede emplear como sintetizador monofónico o bien como teclado de órgano polifónico. Para ello se necesita la siguiente:
- 1 COMMODORE 64

- Un monitor o televisión
- Y FLOPPY DISK o cassette

### EL SINTETIZADOR MONOFONICO

Incluye los siguientes instrumentos:

- |                |                           |
|----------------|---------------------------|
| 1. Trompeta    | 7. Instrumentos de viento |
| 2. Bajo        | 8. Piano                  |
| 3. Clarinete   | 9. El acordeón            |
| 4. Campanas    | 10. Organos de viento     |
| 5. Guitara     | 11. Guitarra              |
| 6. Violonchelo | 12. Batería               |

### Modificaciones paramétricas

- 11. Aumenta el parámetro siguiente
- 12. Regresa al parámetro precedente
- 13. Aumenta el valor del parámetro
- 17. Destruye el valor del parámetro

### EL TECLADO DEL ORGANO POLIFONO

Incluye los siguientes:

- 1. Synchro
- 2. Clave
- 3. Voz de órgano
- 4. Voz de órgano

### Modificaciones paramétricas

- |                  |                   |
|------------------|-------------------|
| 1. Voz de órgano | 12. Voz de órgano |
| 2. Clave         | 13. Voz de órgano |
| 3. Voz de órgano | 14. Voz de órgano |
| 4. Voz de órgano | 15. Voz de órgano |

**¡POR FIN  
LLEGO!**

**A TODO RITMO**

La Caja de Ritmos con todo tipo de variaciones rítmicas aplicables a todos los instrumentos del teclado Music-64



**Albareda**

INSTRUMENTOS  
Y ACCESORIOS MUSICALES

C/. Carmen, 19  
TARREGA (Lérida) · Teléfonos (973) 31 04 02 31 23 51



TABLE 4.1 *Interfaz de fondo del C-64*

Direcciones de Memoria del SID				Chip del SID							
S: 54272											
Var 1	Var 2	Var 3	Nombre de Registro	128	64	32	16	8	4	2	1
S: 5-7	S: 14	S: 15	Baja Frecuencia	F <sub>1</sub>	F <sub>2</sub>	F <sub>3</sub>	F <sub>4</sub>	F <sub>5</sub>	F <sub>6</sub>	F <sub>7</sub>	F <sub>8</sub>
S: 1	S: 8	S: 15	Alta Frecuencia	F <sub>12</sub>	F <sub>11</sub>	F <sub>10</sub>	F <sub>9</sub>	F <sub>8</sub>	F <sub>7</sub>	F <sub>6</sub>	F <sub>5</sub>
S: 1	S: 9	S: 16	Ancho de Pulso - Bajo	PW <sub>1</sub>	PW <sub>2</sub>	PW <sub>3</sub>	PW <sub>4</sub>	PW <sub>5</sub>	PW <sub>6</sub>	PW <sub>7</sub>	PW <sub>8</sub>
S: 4	S: 18	S: 17	Ancho de Pulso - Alto	—	—	—	—	PW <sub>9</sub>	PW <sub>10</sub>	PW <sub>11</sub>	PW <sub>12</sub>
S: 4	S: 11	S: 18	Control de Registro	Rondo	Pulse	Donde de Norma	Triángulo	Test	Mod. de Esc.	Libre	Paralelo
S: 9	S: 12	S: 19	Ataque/Decaimiento (A/D)	A <sub>1</sub>	A <sub>2</sub>	A <sub>3</sub>	A <sub>4</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>	D <sub>4</sub>
S: 8	S: 11	S: 20	Sostenimiento/Relajación (S/R)	S <sub>1</sub>	S <sub>2</sub>	S <sub>3</sub>	S <sub>4</sub>	R <sub>1</sub>	R <sub>2</sub>	R <sub>3</sub>	R <sub>4</sub>
Filtros:	S: 21	S: 22		—	—	—	—	—	F <sub>1</sub>	F <sub>2</sub>	F <sub>3</sub>
	S: 22	S: 23	Resonancia	Res <sub>1</sub>	Res <sub>2</sub>	Res <sub>3</sub>	Res <sub>4</sub>	F <sub>4</sub>	F <sub>5</sub>	F <sub>6</sub>	F <sub>7</sub>
	S: 24	S: 25	Volúmen y Filtros	Vol	HP	BP	LP	Vol <sub>1</sub>	Vol <sub>2</sub>	Vol <sub>3</sub>	Vol <sub>4</sub>
	S: 25	S: 26	Vol 1	PV <sub>1</sub>	PV <sub>2</sub>	PV <sub>3</sub>	PV <sub>4</sub>	PV <sub>5</sub>	PV <sub>6</sub>	PV <sub>7</sub>	PV <sub>8</sub>
	S: 26	S: 27	Vol 2	PV <sub>1</sub>	PV <sub>2</sub>	PV <sub>3</sub>	PV <sub>4</sub>	PV <sub>5</sub>	PV <sub>6</sub>	PV <sub>7</sub>	PV <sub>8</sub>
	S: 27	S: 28	Vol 3	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>3</sub>	Q <sub>4</sub>	Q <sub>5</sub>	Q <sub>6</sub>	Q <sub>7</sub>	Q <sub>8</sub>
	S: 28	S: 29	Envolueta	E <sub>1</sub>	E <sub>2</sub>	E <sub>3</sub>	E <sub>4</sub>	E <sub>5</sub>	E <sub>6</sub>	E <sub>7</sub>	E <sub>8</sub>

TABLE 4.2 *Números de las bits y posiciones correspondientes de 2 con sus equivalencias decimales*

7	6	5	4	3	2	1	0	Nombre de Bit
2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>	Posiciones de 2
128	64	32	16	8	4	2	1	Valores Decimales

TABLE 4.2 *Posiciones de Memoria de fondo en el VM*

Posición de Memoria	Nombre de Registro
36874	Var 1 (rango de 1 a 12)
36875	Var 2 (rango de 1 a 12)
36876	Var 3 (rango de 1 a 12)
36877	Ruido blanco
36878	Control de volumen

## Términos que deberías saber

- Las direcciones del SID, o registros, empiezan en la posición de memoria 54272, que será representada por la constante S. Para referirse a los otros registros del SID, añadiré un número a S, de la siguiente forma: S+4, que significa 54276. A tu vez, generaré el C-64 se llama una vez.
- La nota, o frecuencia, es la alta o la baja de una melodía en la escala musical, que queda determinada por el número de vibraciones de una onda de sonido durante cada ciclo. El chip SID controla la frecuencia de un sonido utilizando dos registros por vez. Estos registros se llaman de baja frecuencia y de alta frecuencia (S y S+1 para la voz).
- La dinámica es la amplitud de sonido; es decir, el volumen. El control del volumen del C-64 (que afecta a todas las voces de la misma forma) se encuentra en S+14, que también controla los filtros. Tiene que contener un número mayor que 0, o de lo contrario no se produce ningún sonido.
- El contenido de armónicos de una forma de onda puede ser cambiado usando un filtro, que normalmente reduce el volumen de la voz. Esto se usa generalmente para aminorar una voz resaltando otra. Hay que indicarle al ordenador la voz que se tiene que filtrar y hasta qué grado, utilizando el registro S+23, y fijar los puntos de referencia en S+21 y S+22 (para la voz 1 solamente).
- El timbre, o color del tono, se refiere a las cualidades individuales por las que se distingue una voz de otra. Por ejemplo, CONCERT A es una frecuencia específica (440 Hz), que constituye un estándar para afinar los instrumentos de una orquesta. Cada músico afinó su instrumento de acuerdo con CONCERT A, nota dada por el primer violín. El timbre es la diferencia que se percibe cuando un arpa y una trompeta tocan la misma nota. Es lo que hace que nuestras voces sean diferentes.
- El control del timbre en el C-64 se realiza mediante los generadores de la envolvente ADSR (dos para cada voz, S+5 y S+6 para la voz 1) que trabajan junto con la forma de onda elegida. ADSR significa Ataque, Decaimiento, Sostenimiento y Relajación. Estas son las características físicas de la envolvente de un sonido.
- La instrumentación se describe como textura. Esto puede ser desde un solo de guitarra hasta una orquesta sinfónica. Lo mismo que puedes tener en el C-64 es tres voces sincronizadas, pero puedes engañar los sentidos para que piensen que se oyen más que eso. Se puede empezar, por ejemplo, con un trío de batería, guitarra, y bajo, pero convertir la guitarra en una trompeta. De esta forma piensas que estás oyendo cuatro voces. Esto se consigue con un programa que coordina voces múltiples.
- El compás incluye el tempo general de la pieza —rápido, despacio o entre medias— y el ritmo.



# Descubre las nuevas impresoras **STAR** para tu Commodore



Star ha lanzado al mercado los nuevos modelos de impresoras para Commodore; doble interface, paralelo centronics y Commodore.

Con este interface para Commodore logramos poder trabajar con todas las grandes prestaciones de esta gama de impresoras, es decir, una gran velocidad, fricción tracción, cinta de maquina de escribir, 100 tipos de letras diferentes... Además de todas estas prestaciones lograrás con tu Commodore transcribir todos estos tipos de letra con sus propios caracteres gráficos.

Si quieres hacer gráficos trabajando con el Simon's Basic, lograrás hacer Hard cyps directamente de pantalla.

Todo esto y mucho más lograrás con las impresoras STAR.

De venta en establecimientos especializados:

IMPORTADO POR:



# SCS

COMPONENTES ELECTRÓNICOS S.A.

08005 BARCELONA: Consejo de Centro, 405  
Tel: (93) 221 59 13

28020 MADRID: Condamante Zorra, 17  
Tel: (91) 233 00 94 - 233 09 24

**M**edrado el tiempo. El programa "C-64MUSIC" se añade al conjunto de tareas de gestión interna por lo que se ejecuta 80 veces cada segundo. Controla el espacio de las notas mediante dos buffers: tiempo y datos.

Si el tiempo es 4 y la duración 3, la nota sonará durante 12 interrupciones, es decir, 15 de segundos. En ese momento, en cuanto de esa voz se desactiva, se carga la siguiente frecuencia en el SID y los circuitos vuelven a funcionar. Como caso se realiza con la velocidad del lenguaje máquina, no se agotan ningún tiempo muerto y el programa Basic se ejecuta a la misma velocidad que si no hubiera música.

El programa que se muestra en el Listado 1 utiliza el lenguaje máquina en memoria y lo salva como un fichero de programa en disco o cinta. El programa reside en un bloque de memoria situado encima del Basic, por lo que no se reduce la cantidad de memoria disponible para el programa Basic.

El Listado 2 es un programa en Basic que se utiliza para preparar los datos para las melodías. El programa puede utilizarse tanto para introducir nuevas melodías como para modificar las ya existentes. Cuando se están modificando, se imprimen los valores actuales en la línea de entrada. De esta forma, si quieres cambiarlos como están, basta pulsar la tecla Return para reintroducirlos.

Cuando se introduce una melodía nueva, primero el programa pide la envolvente ADSR (Ataque, decaimiento, sostenimiento, relajación) de la primera voz. Se pueden introducir valores entre 0 y 15 para cada uno de los parámetros. Además se puede elegir la forma de onda: 1 para ondas triangulares, 2 para ondas de sierra, 3 para pulsos o 4 para ruido. A continuación, el programa se pregunta si prefieres modificaciones en el timbre o sincronización y, si se elige el timbre, pulsas la tecla de entrada del pulso.

Una vez introducidos los parámetros musicales de la voz, se piden los valores correspondientes a las notas. Antes de introducir las notas, se puede introducir una clave. Una vez introducida la clave, no es preciso introducir los sostenidos y bemoles habituales de dicha clave.

Las notas se introducen mediante las letras A-G (A a SOL), el número de octava (de 1 a 7) y su duración. Los sostenidos se indican mediante el símbolo #, mientras que los bemoles han de introducirse mediante su equivalente natural en términos de sostenidos. En la pantalla aparecen los valores de duración para varias notas.

Además de la tecla A-G, se puede introducir una R con un valor 0 de octava para el silencio. También se puede introducir C, V, H, I para cambiar la envolvente ADSR y la forma de onda de la voz durante la ejecución de la melodía. Al final de la melodía se intro-

# Serenata SID

**M**uchos de los juegos existentes para el C-64 son divertidos gracias a la excelente música de fondo que tienen, o a los efectos de sonido que, con frecuencia les acompañan.

Cuando los juegos están escritos en Basic no suelen tener música, ya que es bastante difícil de generar y reduce enormemente la velocidad del desarrollo del juego. Aunque la mayoría de los tareas necesarias para la generación del sonido las realiza el chip SID, el programa no sólo tiene que enviarle los cambios de notas, sino que debe controlar la duración de cada una de ellas.

En este artículo se describe un programa en lenguaje máquina que controla los cambios de notas. Su funcionamiento está basado en la estructura de las interrupciones del 64. La CPU es "interrumpida" (deja de ejecutar la tarea que está realizando y ejecuta algunas tareas de gestión interna antes de proseguir lo que estuviera haciendo) sesenta veces por segundo. Estas tareas incluyen la actualización del reloj TTS y la comprobación del teclado.

M. J. CLIFFORD (BUN DE UJ)

adaptado por Álvaro IBÁÑEZ

Jacar R3, seguido por el número de veces que se desea repetir la melodía (de 0 a 15) y el intervalo que se desea entre las repeticiones.

Cuando se introduce R1, el programa muestra la duración total de la melodía en términos de semibreves y de pausas que se habían introducido. Esto sirve para asegurarse de que las tres voces van sincronizadas. Cuando se termina con la primera voz, se introduce la segunda y luego la tercera.

## Las opciones son las

Una vez introducidos los datos, el programa entra en modo modificación, ofreciendo varias opciones. La melodía puede ejecutarse con tempos distintos, cambiando los envolventes musicales, las notas modificadas o los filtros activados.

Una vez realizadas las modificaciones oportunas, la melodía se salva, junto con el programa, en lenguaje máquina que la interpreta, en un fichero de programa. La melodía se puede cargar posteriormente mediante LOAD "melod-

dia", R1 e interpretarla con un comando SYS.

Una vez que comienza la melodía, la ejecución continuará hasta su conclusión mientras que el Basic funciona normalmente. Hay otro comando SYS que sirve para parar la interpretación antes de su terminación, desconectando al "COD CANC" de las rutinas de interrupción y apagando el SID.

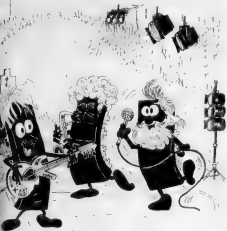
Mediante SYS 49152, la melodía se interrumpe con un tempo (por defecto, de 4). Si quieres volver a un tempo diferente, utiliza el comando POK 780, tempo, SYS 49156. El comando que sirve para detener la ejecución es SYS 49344.

Para ayudarte a comentar, el Listado 3 contiene los datos de una melodía. Ejecuta este programa después de haber ejecutado el cargador, luego carga y ejecuta el Editor de Melodías del Listado 2. Elige el modo Modificación y usa las distintas opciones para ver cómo se introduce la melodía.

Por ejemplo, si eliges la opción 1 (Cambiar la Onda), aparecerán simultáneamente en pantalla los valores de las envolventes ADSR y de las formas de onda. Para volver a introducir el valor sin cambios y pasar al siguiente







prueba la tecla Return según vaya apareciendo cada valor.

Para interpretar la melodía, elige la opción 4. El tiempo apropiado es 6, pero puedes probar con otros valores para comprobar su efecto.

Mediante la opción 5, puedes salvar la melodía junto con el programa en un único archivo, con lo que podrás cargarla posteriormente cuando quieras. El menú 3 ha sido salvado el programa y la melodía, teada 80 N, elige el modo Modificación y experimenta con distintas formas de onda y filtros. Para realizar cambios, teada dos nuevos valores y, a continuación, praba la tecla Return.

#### Descripción del cargador Basic

Linias 10-60. Lee los datos de los líneas 1001-1048 y los coloca en memoria. El primer número de cada línea es de composición. Si se produce algún error en la introducción de los datos, el programa se parará e informará del número de la línea de datos en la que se ha producido el error.

Linias 80-110. Informan que la carga de programa se ha realizado satisfactoriamente y ofrece la posibilidad de sal-

#### Añade música a tus programas en Basic sin perder el ritmo

varlo como un fichero de programa.

Linias 200. Asigna el nombre del fichero de programa.

Linias 210-300. Define la rutina Basic para salvar el programa en disco. Si se va a utilizar cassette, el 6 que aparece en la línea 260 deberá cambiarse a 1.

Nota: Se pueden crear las líneas 95-100 si el programa se ejecuta mediante terminal antes de ejecutar el "Editor de Melodías" (línea 21, ya que este salva el programa en lenguaje máquina junto con los datos. Si decides utilizar en el futuro este procedimiento, asegúrate de que cargas el fichero de la melodía antes de cargar y ejecutar el "SID EDITOR".

#### Descripción del SID EDITOR

Linia 10. Carga el programa en lenguaje máquina si es necesario.

Linias 20-55. Inicializa las variables y permite optar entre "nueva canción" o "editar".

Linias 60-90. Limpia la memoria para albergar nuevas canciones y permite volver a 15 por defecto.

Linias 90-100. Introducción de una nueva melodía.

Linias 110-195. Entrada de la nota octava y duración.

Linias 205-415. Decodifica el valor actual almacenando la nota y octava correspondiente y la muestra junto con la duración en la línea de entrada de forma que pueda ser introducida y guardando la tecla Return.

Linia 420. Entrada de valores.

Linias 425-435. Comparación del final, silencio o cambio de ADNR.

Linias 440-450. Codifica una nota y una octava en un solo byte.

Linia 470. Imprime el tal de semáforos para mantener las tres voces sincronizadas.

Linias 500-790. Junta los valores actuales y los escribe en la línea de entrada por si quieres algún silencio.

Linias 790-900. Entrada de ataque y caída (AD). Los valores se colocan en memoria.

Linias 910-930. Sincronización y relación (NR).

Linias 940-1000. Decodifica la forma de onda actual. 80 N se se que en modulación en audio 80 N se se se desea sin modulación.

Linias 1010-1030. Entrada de la onda, modulación en audio y sin modulación.

Linias 1040-1070. Se se cifra muestra en pulsos, forma el arco del pulso.

Linias 1080-1100. Menú de opciones para modulación.

Linias 1000-1045. Interpreta la melodía.

Linias 1200-1400. No se que nada hasta que el vector de interrupción vuelva a su estado normal cuando termina la melodía, así que se pasa a tecla Q, lo que interrumpe la interpretación de la melodía.

Linias 1500-1700. El subprograma de volámen y de los filtros. Si no se utiliza esta opción, el valor por defecto para el volumen sea 15.

Linias 1800-1895. Inicializa la tabla de valores de las notas según a clave asignada. Los datos son para clave C, pero los sostenidos y bemoles del "Se se" lleva una clave distinta no hay que introducir los bemoles y sostenidos normales para dicha clave.

Linias 1900-1940. Imprime la nota asignada y los valores de octava y duración para varias notas.

Linias 1950-2000. En modulación muestra los sostenidos y bemoles de se clave en caso y permite cambios.

Linias 1900-1930. Final del diálogo.

Linias 2000 y siguientes. Salvo la melodía y el programa en lenguaje máquina como tal como en fichero de programa.

# Starp

## STX - 80

### La silenciosa

Con una velocidad de 60 c.p.s. en silencio, Standard ASCII, con caracteres internacionales programables, bloc gráfico y bit image, es la STX - 80. Con un precio tentador muy fácil de instalar y de programar.

## Gemini 10X

### La estrella

Nº 1 en el ranking de las impresoras. Con 120 c.p.s. Realmente económica y fiable, ahora compatible con IBM: Gemini 10 XI. Para descubrir más cualidades, prueba tu mismo y las conocerás.

## Gemini 11 X

### La económica

La impresora para el profesional con un precio más que razonable. Con una vida del cabezal de más de 100 millones de caracteres. La mejor relación precio-prestaciones.

## Delta 10

### La polivalente

Interface serie y paralelo. Con un buffer de 8K, Microelectrónica. Caracteres programables. Impresión bidireccional y J&K. Otras características estándar. Esto es algo de Delta 10.



# Parade

**SCS**

COMPONENTES ELECTRONICOS, S.A.

CONSEJO DE CIENTO, 409 - 08009 BARCELONA  
TEL. 231 59 13COMANDANTE ZORITA, 13 - 28020 MADRID  
TELS. 233 00 94 - 233 09 24

## Delta 15

### la potencia ampliada

La potencia con carro ancho, es la impresora Delta 15. Mas de 233 columnas con 960 c.p.s. la justa fuerza. Un software inteligente para un precio standard.

## Radix 10

### la nueva profesional

200 c.p.s. con entrada automática de hoja. Un buffer de 16K. Doble velocidad que le permite una letra de calidad. El profesional usa Radix 10. Todas sus prestaciones son de standard.

## Radix 15

### la nueva profesional

#### ampliada

Superior a lo normal. Con una prestaciones que le adelanta al futuro. Doble interfase. Prevista para resolver todos tus problemas. Esto es Radix 15.

## Zowertype

### la nueva calidad de

#### impresión para tu

#### escribitoria.

Con una nueva dimensión a tu correspondencia. Con mas de 100 modelos de margaritas. Una velocidad de 18 c.p.s. Interfase serie y paralelo incluidos. Impresión como la de una máquina de escribir superior. Todo esto a un bajo coste y con una gran calidad.

**star**The power behind  
the printed word.



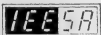
52 190 127 97 111 172 126 189 2062  
1047 0A7014H 264 252 161 195 140 254  
174 227 88 52 129 47 89 247 31 2311

1044 0A7021H 25 252 153 189 176 189  
0 0 0 0 0 0 0 15 1189  
1050 END

#### SERENATA SID SID-EDITOR

```
1 40H SID EDITOR
2 40H SID EDITOR
3 40H SID EDITOR
4 40H SID EDITOR
5 40H SID EDITOR
6 40H SID EDITOR
7 40H SID EDITOR
8 40H SID EDITOR
9 40H SID EDITOR
10 40H SID EDITOR
11 40H SID EDITOR
12 40H SID EDITOR
13 40H SID EDITOR
14 40H SID EDITOR
15 40H SID EDITOR
16 40H SID EDITOR
17 40H SID EDITOR
18 40H SID EDITOR
19 40H SID EDITOR
20 40H SID EDITOR
21 40H SID EDITOR
22 40H SID EDITOR
23 40H SID EDITOR
24 40H SID EDITOR
25 40H SID EDITOR
26 40H SID EDITOR
27 40H SID EDITOR
28 40H SID EDITOR
29 40H SID EDITOR
30 40H SID EDITOR
31 40H SID EDITOR
32 40H SID EDITOR
33 40H SID EDITOR
34 40H SID EDITOR
35 40H SID EDITOR
36 40H SID EDITOR
37 40H SID EDITOR
38 40H SID EDITOR
39 40H SID EDITOR
40 40H SID EDITOR
41 40H SID EDITOR
42 40H SID EDITOR
43 40H SID EDITOR
44 40H SID EDITOR
45 40H SID EDITOR
46 40H SID EDITOR
47 40H SID EDITOR
48 40H SID EDITOR
49 40H SID EDITOR
50 40H SID EDITOR
51 40H SID EDITOR
52 40H SID EDITOR
53 40H SID EDITOR
54 40H SID EDITOR
55 40H SID EDITOR
56 40H SID EDITOR
57 40H SID EDITOR
58 40H SID EDITOR
59 40H SID EDITOR
60 40H SID EDITOR
61 40H SID EDITOR
62 40H SID EDITOR
63 40H SID EDITOR
64 40H SID EDITOR
65 40H SID EDITOR
66 40H SID EDITOR
67 40H SID EDITOR
68 40H SID EDITOR
69 40H SID EDITOR
70 40H SID EDITOR
71 40H SID EDITOR
72 40H SID EDITOR
73 40H SID EDITOR
74 40H SID EDITOR
75 40H SID EDITOR
76 40H SID EDITOR
77 40H SID EDITOR
78 40H SID EDITOR
79 40H SID EDITOR
80 40H SID EDITOR
81 40H SID EDITOR
82 40H SID EDITOR
83 40H SID EDITOR
84 40H SID EDITOR
85 40H SID EDITOR
86 40H SID EDITOR
87 40H SID EDITOR
88 40H SID EDITOR
89 40H SID EDITOR
90 40H SID EDITOR
91 40H SID EDITOR
92 40H SID EDITOR
93 40H SID EDITOR
94 40H SID EDITOR
95 40H SID EDITOR
96 40H SID EDITOR
97 40H SID EDITOR
98 40H SID EDITOR
99 40H SID EDITOR
100 40H SID EDITOR
```

```
70 40H SID EDITOR
71 40H SID EDITOR
72 40H SID EDITOR
73 40H SID EDITOR
74 40H SID EDITOR
75 40H SID EDITOR
76 40H SID EDITOR
77 40H SID EDITOR
78 40H SID EDITOR
79 40H SID EDITOR
80 40H SID EDITOR
81 40H SID EDITOR
82 40H SID EDITOR
83 40H SID EDITOR
84 40H SID EDITOR
85 40H SID EDITOR
86 40H SID EDITOR
87 40H SID EDITOR
88 40H SID EDITOR
89 40H SID EDITOR
90 40H SID EDITOR
91 40H SID EDITOR
92 40H SID EDITOR
93 40H SID EDITOR
94 40H SID EDITOR
95 40H SID EDITOR
96 40H SID EDITOR
97 40H SID EDITOR
98 40H SID EDITOR
99 40H SID EDITOR
100 40H SID EDITOR
```



C/ MIGUEL YUSTE, 16 - TEL.: 204 51 98, MADRID

## COMMODORE

ORDENADORES - PERIFERICOS - SOFTWARE

## SERVICIO TECNICO

ENVIOS A PROVINCIAS

SI NO LO ENCUENTRA EN MICROTERSA OLVIDELO









**P.V.P. 2.950 Ptas.**

*Precio adjuntando boleto: 2.500 Ptas.*



**REGLA  
RELOJ CALCULADORA**

- *Relej con posición "vertical" para más fácil lectura.*
- *Calculadora "extrañble" de la regla con las 4 funciones elementales (+, -, x, ÷, posee raíz y % así como memoria.*
- *Tabla de conversión de medidas impresa en la regla.*

**GRAN PENZA**

*(en madera barnizada)  
Sirve tanto de pizapapeles como  
para sujetarlos, sirve imprimos los  
distintivos de todas nuestras  
publicaciones.*



**P.V.P. 400 Ptas.**

*Precio adjuntando boleto: 300 Ptas.*

# BOUTIQUE Commodore World

Si eres "commodoriano"  
... ¡¡¡Que lo sepan!!!

**CAMISETA DE FELPA**

- *Estampada en el anverso con el  
distintivo de Commodore World y en  
el reverso con el distintivo de las  
publicaciones hermanas,  
MicroSistemas y PC World.*

**P.V.P. 1.950 Ptas.**

*Precio adjuntando boleto: 1.500 Ptas.*

**BOUTIQUE Commodore World - Boleto de Pedido**

NOMBRE \_\_\_\_\_

DIRECCION \_\_\_\_\_

POBLACION \_\_\_\_\_

(C.P. ) PROYECTA \_\_\_\_\_

TELEF. \_\_\_\_\_

FORMA DE PAGO

- ☐ Cheque por valor de \_\_\_\_\_  
☐ Efecto giro nº \_\_\_\_\_ por \_\_\_\_\_

DESEO ME ENVIE LA CANTIDAD DE:

- ☐ REGLAS RELOJ-CALCULADORAS a 1.900 Ptas. c/u  
☐ CAMISETAS DE FELPA a 1.900 Ptas. c/u  
TALLAS MEDIANAS O GRANDE  
☐ GRAN PENZA a 300 Ptas. c/u.

\_\_\_\_\_ pesetas  
\_\_\_\_\_ pesetas + 100 Ptas. por gastos de envío por unidad.

ENVIAR A COMMODORE WORLD • C/ BARCELLO, 21-P IZQDA. • 28006 MADRID

# Volando con el VIC

A ver si eres capaz de despegar y volar tu avión a través del cañón sin chocar con las paredes ni con los aviones que vienen en dirección contraria.

**V**olando con el VIC es un juego tipo "arcade", para jugar con joystick en el VIC-20 no ampliado. Tienes que despegar y volar a través del cañón sin chocar con las paredes ni con los otros aviones. Si llegas a los 2.000 puntos, el mensaje "LO HICISTE" aparece en pantalla. Hay cuatro niveles de dificultad. Van desde un juego corto (C) y posible (P) hasta un juego largo (L) e imposible (I). No comienza a nadie que haya llegado al nivel más difícil, dado su forma en que está programado el juego.

## VIC-20 NO AMPLIADO CINTA O DISCO

Juego dedicado al aeropuerto, una subrutina localizada al final del programa, consiste en la simulación, las sentencias que dibujan el aeropuerto y las maniobras del avión. La siguiente sección comienza la simulación del cañón, el hecho de volar por el cañón, y las rutinas de choque y explosión y programación.



Para cargar los programas, tecla primero el Estado 1. Se van a cargar el programa en disco, entre la línea 130 y 140, y para las sentencias RLM de las líneas 140 y 150.

En cualquier momento tecla la sentencia Data, ya que contienen los caracteres personalizados de las rutinas del aeropuerto y del cañón. Salva el programa antes de ejecutarlo.

Segundo, tecla el Estado 2, y el valor con el nombre "I 0" y estás volando la ciudad de discos. Si estás utilizando cinta, salva el Estado 2 directamente después del Estado 1. De esa forma, cuando se ejecuta el Estado 1, el Estado 2 se carga y se ejecuta automáticamente.

A continuación se presenta una descripción más detallada del Estado 2 y del modo del juego.

### Iniciación

Los parámetros necesarios para la inicialización se encuentran en tres secciones del programa: al principio, para establecer el nivel de dificultad y el color de la pantalla, y para activar las cata-

### Descripción del programa

El programa está escrito en Basic, prestando mucha atención a su estructura al hacer principal del juego para que sea lo más rápido posible. Dada la memoria limitada del VIC no ampliado, se carga el programa en dos partes.

El Estado 1 presenta las instrucciones e introduce en la memoria, mediante unos pokes, los datos para los 60 caracteres personalizados. Finalmente, los 512 bytes de memoria de caracteres personalizados están protegidos, y el Estado 2, el programa principal, está cargado y ejecutado mediante el uso de unos pokes en el buffer del teclado.

El Estado 2 comienza en tres secciones principales: la simulación, el aeropuerto y las maniobras en el cañón. La parte del



tenes personalidades, en la subrutina del aeropuerto, para establecer las sentencias que dibujan los caracteres personalizados para crear el aeropuerto, y luego para determinar el sonido, las constantes del joystick y la posición inicial del avión, en la rutina del cañón, para visualizar las dimensiones de los "arrays", las "strings" y la posición inicial de la entrada del cañón y el avión.

En la subrutina del aeropuerto, el avión aparece en pantalla, se detecta la dirección del joystick, y el avión se mueve de acuerdo con esto. Se incluyen sentencias IF. Then para determinar si el avión se encuentra dentro de los límites de la pantalla, si ha chocado con cualquier cosa en el aeropuerto y, finalmente, si se dirige al sur hacia el cañón. Cuando se cumple esta última condición, la parte del programa que incluye el movimiento por el cañón asume el control del avión. Como verás ahora, el bucle del cañón está programado aparte para que sea más rápido.

#### El cañón

Para empezar, se realizan unas operaciones de "mantenimiento", incluyendo fijar las "strings" (líneas 100-140) que se utilizan para crear las sentencias aleatorias que determinan la dirección en que se gira el cañón. Toma nota que a diferencia de otros programas de gráficos parecidos que utilizan sentencias de este tipo, en cuatro "strings" ES se crean utilizando caracteres personalizados iguales en los bordes del cañón.

Las "strings" ES se digitan utilizando el "array" L1(2), lo que permite que el cañón tenga esos bordes suaves a medida que pasan las secciones hacia la derecha, la izquierda o recto. La línea 150 utiliza dos sentencias aleatorias —Y, para evaluar la entrada del cañón, y X, para el color del avión—. La sentencia POC de la línea 160 coloca el cursor en la periferia fija de la parte inferior de la pantalla. Aquí se encuentra la primera sentencia del cañón.

Las líneas 170-230 construyen el cañón del juego. Estas forman el bucle que controla el avión, dibuja el cañón y detecta los choques. En estas líneas se han tomado medidas para que este programa en Basic vaya más rápido. Por

## ***E***l programa no es largo y se presta a modificaciones.

**Los "strings" en las líneas 100-140 que dibujan el cañón se componen en su mayoría de caracteres gráficos elegidos al azar. A medida que juegas, observa lo que pasa con estas sentencias.**

ejemplo, un bucle For...Next ha sido empleado en las líneas 170 y 270, esto resulta más rápido que usar GOTO 170 en la línea 270.

Se ha empleado el comando RND(0) en vez del comando más normal RND(1), y se han usado puntos (.) para sustituir los ceros 00 en las líneas 170, 180, 190, 210, 240, 250 y 260; todo esto aumenta la velocidad del Basic.

Las sentencias IF de las líneas 170 y 180 deciden aleatoriamente si el cañón gira hacia la izquierda, la derecha o sigue recto. Las líneas 190 y 200 comienzan las sentencias IF que mantienen el cañón en pantalla. En la línea 210, el avión baja por la pantalla y se comprueba para ver si llegó a pasar por el cañón. Si quieres un juego más largo o más corto, puedes modificar la variable W en las líneas 20 y 30.

En la línea 220, la primera sentencia calcula Q\$, el "string" que dibuja en pantalla, a continuación el color actual del avión se convierte en el color del fondo mediante unos pokes para preparar el movimiento. El "string" Q\$ se imprime en la parte inferior de la pantalla y todas las otras líneas son empuñadas hacia arriba, dando la impresión de que el avión se está desplazando por el cañón. A continuación, se actualiza la posición del avión y este se convierte en amarillo mediante unos pokes para que vuelva a ser visible. En síntesis, K, asume el valor de X. Se detecta el choque con la pared y con otro avión

mediante la sentencia IF. Then de la línea siguiente.

La línea 240 utiliza los pokes para sacar aviones de colores al azar. La frecuencia con que aparecen estos aviones está controlada por la variable D, fijada por la opción posible/imposible. Puedes cambiar el juego para que resulte más fácil o más difícil si cambias el valor de la variable D en la línea 20.

Las dos líneas siguientes leen el joystick, modifican X, la posición del avión, y fijan P, el carácter personalizado para que el avión vaya hacia la derecha, izquierda o recto. La última línea del bucle es una sencilla Nest y vuelve a entrar el programa por el bucle.

#### Últimas Rutinas y Modificaciones

Las líneas 280 y 290 producen los efectos visuales y sonoros del choque. La pantalla rembla haciendo un poke en la dirección localizada en el centro de la pantalla a la vez que los colores de la pantalla se cambian aleatoriamente. Después de la explosión, se actualiza la puntuación y los comentarios sobre el resultado aparecen en pantalla. El jugador tiene que pulsar luego para volver a jugar.

El programa no es largo y se presta a modificaciones. Los "strings" en las líneas 100-140 que dibujan el cañón se componen en su mayoría de caracteres gráficos elegidos al azar. A medida que juegas, observa lo que pasa con estas sentencias. Verás que cambian los caracteres individuales. Esto es el resultado de sacar los caracteres gráficos de una parte de la memoria que es el programa en Basic, en vez de la ROM de caracteres.

Sin embargo, la entrada al cañón y los caracteres más inmediatos en cada lado son más importantes. Puedes intentar cambiar la anchura del cañón para que el juego resulte más fácil, lo modificas la línea 240, puedes introducir cualquier obstáculo que no tiene un avión.

Si el juego resulta demasiado difícil, puedes fijar la sentencia RND en un valor que no sea >0.4. Si lo haces a >0.87, salen más secciones rectas en el cañón, y resulta mucho más fácil ganar. ¿Que te dierais?

#### LISTADO 1

```
1:CLS:PO
2:FOR I=1 TO 4:ARR=INT(RND*255)
3:ARR=ARR*16:FOR J=0 TO 15:PRINT
4:PRINT:NEXT J:PRINT:NEXT I
5:FOR I=0 TO 15:ARR=INT(RND*16)
6:ARR=ARR*16:PRINT:NEXT I
7:FOR I=0 TO 15:ARR=INT(RND*16)
8:ARR=ARR*16:PRINT:NEXT I
9:FOR I=0 TO 15:ARR=INT(RND*16)
10:ARR=ARR*16:PRINT:NEXT I
11:FOR I=0 TO 15:ARR=INT(RND*16)
12:ARR=ARR*16:PRINT:NEXT I
13:FOR I=0 TO 15:ARR=INT(RND*16)
14:ARR=ARR*16:PRINT:NEXT I
15:FOR I=0 TO 15:ARR=INT(RND*16)
16:ARR=ARR*16:PRINT:NEXT I
17:FOR I=0 TO 15:ARR=INT(RND*16)
18:ARR=ARR*16:PRINT:NEXT I
19:FOR I=0 TO 15:ARR=INT(RND*16)
20:ARR=ARR*16:PRINT:NEXT I
21:FOR I=0 TO 15:ARR=INT(RND*16)
22:ARR=ARR*16:PRINT:NEXT I
23:FOR I=0 TO 15:ARR=INT(RND*16)
24:ARR=ARR*16:PRINT:NEXT I
25:FOR I=0 TO 15:ARR=INT(RND*16)
26:ARR=ARR*16:PRINT:NEXT I
27:FOR I=0 TO 15:ARR=INT(RND*16)
28:ARR=ARR*16:PRINT:NEXT I
29:FOR I=0 TO 15:ARR=INT(RND*16)
30:ARR=ARR*16:PRINT:NEXT I
31:FOR I=0 TO 15:ARR=INT(RND*16)
32:ARR=ARR*16:PRINT:NEXT I
33:FOR I=0 TO 15:ARR=INT(RND*16)
34:ARR=ARR*16:PRINT:NEXT I
35:FOR I=0 TO 15:ARR=INT(RND*16)
36:ARR=ARR*16:PRINT:NEXT I
37:FOR I=0 TO 15:ARR=INT(RND*16)
38:ARR=ARR*16:PRINT:NEXT I
39:FOR I=0 TO 15:ARR=INT(RND*16)
40:ARR=ARR*16:PRINT:NEXT I
41:FOR I=0 TO 15:ARR=INT(RND*16)
42:ARR=ARR*16:PRINT:NEXT I
43:FOR I=0 TO 15:ARR=INT(RND*16)
44:ARR=ARR*16:PRINT:NEXT I
45:FOR I=0 TO 15:ARR=INT(RND*16)
46:ARR=ARR*16:PRINT:NEXT I
47:FOR I=0 TO 15:ARR=INT(RND*16)
48:ARR=ARR*16:PRINT:NEXT I
49:FOR I=0 TO 15:ARR=INT(RND*16)
50:ARR=ARR*16:PRINT:NEXT I
51:FOR I=0 TO 15:ARR=INT(RND*16)
52:ARR=ARR*16:PRINT:NEXT I
53:FOR I=0 TO 15:ARR=INT(RND*16)
54:ARR=ARR*16:PRINT:NEXT I
55:FOR I=0 TO 15:ARR=INT(RND*16)
56:ARR=ARR*16:PRINT:NEXT I
57:FOR I=0 TO 15:ARR=INT(RND*16)
58:ARR=ARR*16:PRINT:NEXT I
59:FOR I=0 TO 15:ARR=INT(RND*16)
60:ARR=ARR*16:PRINT:NEXT I
61:FOR I=0 TO 15:ARR=INT(RND*16)
62:ARR=ARR*16:PRINT:NEXT I
63:FOR I=0 TO 15:ARR=INT(RND*16)
64:ARR=ARR*16:PRINT:NEXT I
65:FOR I=0 TO 15:ARR=INT(RND*16)
66:ARR=ARR*16:PRINT:NEXT I
67:FOR I=0 TO 15:ARR=INT(RND*16)
68:ARR=ARR*16:PRINT:NEXT I
69:FOR I=0 TO 15:ARR=INT(RND*16)
70:ARR=ARR*16:PRINT:NEXT I
71:FOR I=0 TO 15:ARR=INT(RND*16)
72:ARR=ARR*16:PRINT:NEXT I
73:FOR I=0 TO 15:ARR=INT(RND*16)
74:ARR=ARR*16:PRINT:NEXT I
75:FOR I=0 TO 15:ARR=INT(RND*16)
76:ARR=ARR*16:PRINT:NEXT I
77:FOR I=0 TO 15:ARR=INT(RND*16)
78:ARR=ARR*16:PRINT:NEXT I
79:FOR I=0 TO 15:ARR=INT(RND*16)
80:ARR=ARR*16:PRINT:NEXT I
81:FOR I=0 TO 15:ARR=INT(RND*16)
82:ARR=ARR*16:PRINT:NEXT I
83:FOR I=0 TO 15:ARR=INT(RND*16)
84:ARR=ARR*16:PRINT:NEXT I
85:FOR I=0 TO 15:ARR=INT(RND*16)
86:ARR=ARR*16:PRINT:NEXT I
87:FOR I=0 TO 15:ARR=INT(RND*16)
88:ARR=ARR*16:PRINT:NEXT I
89:FOR I=0 TO 15:ARR=INT(RND*16)
90:ARR=ARR*16:PRINT:NEXT I
91:FOR I=0 TO 15:ARR=INT(RND*16)
92:ARR=ARR*16:PRINT:NEXT I
93:FOR I=0 TO 15:ARR=INT(RND*16)
94:ARR=ARR*16:PRINT:NEXT I
95:FOR I=0 TO 15:ARR=INT(RND*16)
96:ARR=ARR*16:PRINT:NEXT I
97:FOR I=0 TO 15:ARR=INT(RND*16)
98:ARR=ARR*16:PRINT:NEXT I
99:FOR I=0 TO 15:ARR=INT(RND*16)
100:ARR=ARR*16:PRINT:NEXT I
101:FOR I=0 TO 15:ARR=INT(RND*16)
102:ARR=ARR*16:PRINT:NEXT I
103:FOR I=0 TO 15:ARR=INT(RND*16)
104:ARR=ARR*16:PRINT:NEXT I
105:FOR I=0 TO 15:ARR=INT(RND*16)
106:ARR=ARR*16:PRINT:NEXT I
107:FOR I=0 TO 15:ARR=INT(RND*16)
108:ARR=ARR*16:PRINT:NEXT I
109:FOR I=0 TO 15:ARR=INT(RND*16)
110:ARR=ARR*16:PRINT:NEXT I
111:FOR I=0 TO 15:ARR=INT(RND*16)
112:ARR=ARR*16:PRINT:NEXT I
113:FOR I=0 TO 15:ARR=INT(RND*16)
114:ARR=ARR*16:PRINT:NEXT I
115:FOR I=0 TO 15:ARR=INT(RND*16)
116:ARR=ARR*16:PRINT:NEXT I
117:FOR I=0 TO 15:ARR=INT(RND*16)
118:ARR=ARR*16:PRINT:NEXT I
119:FOR I=0 TO 15:ARR=INT(RND*16)
120:ARR=ARR*16:PRINT:NEXT I
121:FOR I=0 TO 15:ARR=INT(RND*16)
122:ARR=ARR*16:PRINT:NEXT I
123:FOR I=0 TO 15:ARR=INT(RND*16)
124:ARR=ARR*16:PRINT:NEXT I
125:FOR I=0 TO 15:ARR=INT(RND*16)
126:ARR=ARR*16:PRINT:NEXT I
127:FOR I=0 TO 15:ARR=INT(RND*16)
128:ARR=ARR*16:PRINT:NEXT I
129:FOR I=0 TO 15:ARR=INT(RND*16)
130:ARR=ARR*16:PRINT:NEXT I
131:FOR I=0 TO 15:ARR=INT(RND*16)
132:ARR=ARR*16:PRINT:NEXT I
133:FOR I=0 TO 15:ARR=INT(RND*16)
134:ARR=ARR*16:PRINT:NEXT I
135:FOR I=0 TO 15:ARR=INT(RND*16)
136:ARR=ARR*16:PRINT:NEXT I
137:FOR I=0 TO 15:ARR=INT(RND*16)
138:ARR=ARR*16:PRINT:NEXT I
139:FOR I=0 TO 15:ARR=INT(RND*16)
140:ARR=ARR*16:PRINT:NEXT I
141:FOR I=0 TO 15:ARR=INT(RND*16)
142:ARR=ARR*16:PRINT:NEXT I
143:FOR I=0 TO 15:ARR=INT(RND*16)
144:ARR=ARR*16:PRINT:NEXT I
145:FOR I=0 TO 15:ARR=INT(RND*16)
146:ARR=ARR*16:PRINT:NEXT I
147:FOR I=0 TO 15:ARR=INT(RND*16)
148:ARR=ARR*16:PRINT:NEXT I
149:FOR I=0 TO 15:ARR=INT(RND*16)
150:ARR=ARR*16:PRINT:NEXT I
151:FOR I=0 TO 15:ARR=INT(RND*16)
152:ARR=ARR*16:PRINT:NEXT I
153:FOR I=0 TO 15:ARR=INT(RND*16)
154:ARR=ARR*16:PRINT:NEXT I
155:FOR I=0 TO 15:ARR=INT(RND*16)
156:ARR=ARR*16:PRINT:NEXT I
157:FOR I=0 TO 15:ARR=INT(RND*16)
158:ARR=ARR*16:PRINT:NEXT I
159:FOR I=0 TO 15:ARR=INT(RND*16)
160:ARR=ARR*16:PRINT:NEXT I
161:FOR I=0 TO 15:ARR=INT(RND*16)
162:ARR=ARR*16:PRINT:NEXT I
163:FOR I=0 TO 15:ARR=INT(RND*16)
164:ARR=ARR*16:PRINT:NEXT I
165:FOR I=0 TO 15:ARR=INT(RND*16)
166:ARR=ARR*16:PRINT:NEXT I
167:FOR I=0 TO 15:ARR=INT(RND*16)
168:ARR=ARR*16:PRINT:NEXT I
169:FOR I=0 TO 15:ARR=INT(RND*16)
170:ARR=ARR*16:PRINT:NEXT I
171:FOR I=0 TO 15:ARR=INT(RND*16)
172:ARR=ARR*16:PRINT:NEXT I
173:FOR I=0 TO 15:ARR=INT(RND*16)
174:ARR=ARR*16:PRINT:NEXT I
175:FOR I=0 TO 15:ARR=INT(RND*16)
176:ARR=ARR*16:PRINT:NEXT I
177:FOR I=0 TO 15:ARR=INT(RND*16)
178:ARR=ARR*16:PRINT:NEXT I
179:FOR I=0 TO 15:ARR=INT(RND*16)
180:ARR=ARR*16:PRINT:NEXT I
181:FOR I=0 TO 15:ARR=INT(RND*16)
182:ARR=ARR*16:PRINT:NEXT I
183:FOR I=0 TO 15:ARR=INT(RND*16)
184:ARR=ARR*16:PRINT:NEXT I
185:FOR I=0 TO 15:ARR=INT(RND*16)
186:ARR=ARR*16:PRINT:NEXT I
187:FOR I=0 TO 15:ARR=INT(RND*16)
188:ARR=ARR*16:PRINT:NEXT I
189:FOR I=0 TO 15:ARR=INT(RND*16)
190:ARR=ARR*16:PRINT:NEXT I
191:FOR I=0 TO 15:ARR=INT(RND*16)
192:ARR=ARR*16:PRINT:NEXT I
193:FOR I=0 TO 15:ARR=INT(RND*16)
194:ARR=ARR*16:PRINT:NEXT I
195:FOR I=0 TO 15:ARR=INT(RND*16)
196:ARR=ARR*16:PRINT:NEXT I
197:FOR I=0 TO 15:ARR=INT(RND*16)
198:ARR=ARR*16:PRINT:NEXT I
199:FOR I=0 TO 15:ARR=INT(RND*16)
200:ARR=ARR*16:PRINT:NEXT I
201:FOR I=0 TO 15:ARR=INT(RND*16)
202:ARR=ARR*16:PRINT:NEXT I
203:FOR I=0 TO 15:ARR=INT(RND*16)
204:ARR=ARR*16:PRINT:NEXT I
205:FOR I=0 TO 15:ARR=INT(RND*16)
206:ARR=ARR*16:PRINT:NEXT I
207:FOR I=0 TO 15:ARR=INT(RND*16)
208:ARR=ARR*16:PRINT:NEXT I
209:FOR I=0 TO 15:ARR=INT(RND*16)
210:ARR=ARR*16:PRINT:NEXT I
211:FOR I=0 TO 15:ARR=INT(RND*16)
212:ARR=ARR*16:PRINT:NEXT I
213:FOR I=0 TO 15:ARR=INT(RND*16)
214:ARR=ARR*16:PRINT:NEXT I
215:FOR I=0 TO 15:ARR=INT(RND*16)
216:ARR=ARR*16:PRINT:NEXT I
217:FOR I=0 TO 15:ARR=INT(RND*16)
218:ARR=ARR*16:PRINT:NEXT I
219:FOR I=0 TO 15:ARR=INT(RND*16)
220:ARR=ARR*16:PRINT:NEXT I
221:FOR I=0 TO 15:ARR=INT(RND*16)
222:ARR=ARR*16:PRINT:NEXT I
223:FOR I=0 TO 15:ARR=INT(RND*16)
224:ARR=ARR*16:PRINT:NEXT I
225:FOR I=0 TO 15:ARR=INT(RND*16)
226:ARR=ARR*16:PRINT:NEXT I
227:FOR I=0 TO 15:ARR=INT(RND*16)
228:ARR=ARR*16:PRINT:NEXT I
229:FOR I=0 TO 15:ARR=INT(RND*16)
230:ARR=ARR*16:PRINT:NEXT I
231:FOR I=0 TO 15:ARR=INT(RND*16)
232:ARR=ARR*16:PRINT:NEXT I
233:FOR I=0 TO 15:ARR=INT(RND*16)
234:ARR=ARR*16:PRINT:NEXT I
235:FOR I=0 TO 15:ARR=INT(RND*16)
236:ARR=ARR*16:PRINT:NEXT I
237:FOR I=0 TO 15:ARR=INT(RND*16)
238:ARR=ARR*16:PRINT:NEXT I
239:FOR I=0 TO 15:ARR=INT(RND*16)
240:ARR=ARR*16:PRINT:NEXT I
241:FOR I=0 TO 15:ARR=INT(RND*16)
242:ARR=ARR*16:PRINT:NEXT I
243:FOR I=0 TO 15:ARR=INT(RND*16)
244:ARR=ARR*16:PRINT:NEXT I
245:FOR I=0 TO 15:ARR=INT(RND*16)
246:ARR=ARR*16:PRINT:NEXT I
247:FOR I=0 TO 15:ARR=INT(RND*16)
248:ARR=ARR*16:PRINT:NEXT I
249:FOR I=0 TO 15:ARR=INT(RND*16)
250:ARR=ARR*16:PRINT:NEXT I
251:FOR I=0 TO 15:ARR=INT(RND*16)
252:ARR=ARR*16:PRINT:NEXT I
253:FOR I=0 TO 15:ARR=INT(RND*16)
254:ARR=ARR*16:PRINT:NEXT I
255:FOR I=0 TO 15:ARR=INT(RND*16)
256:ARR=ARR*16:PRINT:NEXT I
257:FOR I=0 TO 15:ARR=INT(RND*16)
258:ARR=ARR*16:PRINT:NEXT I
259:FOR I=0 TO 15:ARR=INT(RND*16)
260:ARR=ARR*16:PRINT:NEXT I
261:FOR I=0 TO 15:ARR=INT(RND*16)
262:ARR=ARR*16:PRINT:NEXT I
263:FOR I=0 TO 15:ARR=INT(RND*16)
264:ARR=ARR*16:PRINT:NEXT I
265:FOR I=0 TO 15:ARR=INT(RND*16)
266:ARR=ARR*16:PRINT:NEXT I
267:FOR I=0 TO 15:ARR=INT(RND*16)
268:ARR=ARR*16:PRINT:NEXT I
269:FOR I=0 TO 15:ARR=INT(RND*16)
270:ARR=ARR*16:PRINT:NEXT I
271:FOR I=0 TO 15:ARR=INT(RND*16)
272:ARR=ARR*16:PRINT:NEXT I
273:FOR I=0 TO 15:ARR=INT(RND*16)
274:ARR=ARR*16:PRINT:NEXT I
275:FOR I=0 TO 15:ARR=INT(RND*16)
276:ARR=ARR*16:PRINT:NEXT I
277:FOR I=0 TO 15:ARR=INT(RND*16)
278:ARR=ARR*16:PRINT:NEXT I
279:FOR I=0 TO 15:ARR=INT(RND*16)
280:ARR=ARR*16:PRINT:NEXT I
281:FOR I=0 TO 15:ARR=INT(RND*16)
282:ARR=ARR*16:PRINT:NEXT I
283:FOR I=0 TO 15:ARR=INT(RND*16)
284:ARR=ARR*16:PRINT:NEXT I
285:FOR I=0 TO 15:ARR=INT(RND*16)
286:ARR=ARR*16:PRINT:NEXT I
287:FOR I=0 TO 15:ARR=INT(RND*16)
288:ARR=ARR*16:PRINT:NEXT I
289:FOR I=0 TO 15:ARR=INT(RND*16)
290:ARR=ARR*16:PRINT:NEXT I
291:FOR I=0 TO 15:ARR=INT(RND*16)
292:ARR=ARR*16:PRINT:NEXT I
293:FOR I=0 TO 15:ARR=INT(RND*16)
294:ARR=ARR*16:PRINT:NEXT I
295:FOR I=0 TO 15:ARR=INT(RND*16)
296:ARR=ARR*16:PRINT:NEXT I
297:FOR I=0 TO 15:ARR=INT(RND*16)
298:ARR=ARR*16:PRINT:NEXT I
299:FOR I=0 TO 15:ARR=INT(RND*16)
300:ARR=ARR*16:PRINT:NEXT I
301:FOR I=0 TO 15:ARR=INT(RND*16)
302:ARR=ARR*16:PRINT:NEXT I
303:FOR I=0 TO 15:ARR=INT(RND*16)
304:ARR=ARR*16:PRINT:NEXT I
305:FOR I=0 TO 15:ARR=INT(RND*16)
306:ARR=ARR*16:PRINT:NEXT I
307:FOR I=0 TO 15:ARR=INT(RND*16)
308:ARR=ARR*16:PRINT:NEXT I
309:FOR I=0 TO 15:ARR=INT(RND*16)
310:ARR=ARR*16:PRINT:NEXT I
311:FOR I=0 TO 15:ARR=INT(RND*16)
312:ARR=ARR*16:PRINT:NEXT I
313:FOR I=0 TO 15:ARR=INT(RND*16)
314:ARR=ARR*16:PRINT:NEXT I
315:FOR I=0 TO 15:ARR=INT(RND*16)
316:ARR=ARR*16:PRINT:NEXT I
317:FOR I=0 TO 15:ARR=INT(RND*16)
318:ARR=ARR*16:PRINT:NEXT I
319:FOR I=0 TO 15:ARR=INT(RND*16)
320:ARR=ARR*16:PRINT:NEXT I
321:FOR I=0 TO 15:ARR=INT(RND*16)
322:ARR=ARR*16:PRINT:NEXT I
323:FOR I=0 TO 15:ARR=INT(RND*16)
324:ARR=ARR*16:PRINT:NEXT I
325:FOR I=0 TO 15:ARR=INT(RND*16)
326:ARR=ARR*16:PRINT:NEXT I
327:FOR I=0 TO 15:ARR=INT(RND*16)
328:ARR=ARR*16:PRINT:NEXT I
329:FOR I=0 TO 15:ARR=INT(RND*16)
330:ARR=ARR*16:PRINT:NEXT I
331:FOR I=0 TO 15:ARR=INT(RND*16)
332:ARR=ARR*16:PRINT:NEXT I
333:FOR I=0 TO 15:ARR=INT(RND*16)
334:ARR=ARR*16:PRINT:NEXT I
335:FOR I=0 TO 15:ARR=INT(RND*16)
336:ARR=ARR*16:PRINT:NEXT I
337:FOR I=0 TO 15:ARR=INT(RND*16)
338:ARR=ARR*16:PRINT:NEXT I
339:FOR I=0 TO 15:ARR=INT(RND*16)
340:ARR=ARR*16:PRINT:NEXT I
341:FOR I=0 TO 15:ARR=INT(RND*16)
342:ARR=ARR*16:PRINT:NEXT I
343:FOR I=0 TO 15:ARR=INT(RND*16)
344:ARR=ARR*16:PRINT:NEXT I
345:FOR I=0 TO 15:ARR=INT(RND*16)
346:ARR=ARR*16:PRINT:NEXT I
347:FOR I=0 TO 15:ARR=INT(RND*16)
348:ARR=ARR*16:PRINT:NEXT I
349:FOR I=0 TO 15:ARR=INT(RND*16)
350:ARR=ARR*16:PRINT:NEXT I
351:FOR I=0 TO 15:ARR=INT(RND*16)
352:ARR=ARR*16:PRINT:NEXT I
353:FOR I=0 TO 15:ARR=INT(RND*16)
354:ARR=ARR*16:PRINT:NEXT I
355:FOR I=0 TO 15:ARR=INT(RND*16)
356:ARR=ARR*16:PRINT:NEXT I
357:FOR I=0 TO 15:ARR=INT(RND*16)
358:ARR=ARR*16:PRINT:NEXT I
359:FOR I=0 TO 15:ARR=INT(RND*16)
360:ARR=ARR*16:PRINT:NEXT I
361:FOR I=0 TO 15:ARR=INT(RND*16)
362:ARR=ARR*16:PRINT:NEXT I
363:FOR I=0 TO 15:ARR=INT(RND*16)
364:ARR=ARR*16:PRINT:NEXT I
365:FOR I=0 TO 15:ARR=INT(RND*16)
366:ARR=ARR*16:PRINT:NEXT I
367:FOR I=0 TO 15:ARR=INT(RND*16)
368:ARR=ARR*16:PRINT:NEXT I
369:FOR I=0 TO 15:ARR=INT(RND*16)
370:ARR=ARR*16:PRINT:NEXT I
371:FOR I=0 TO 15:ARR=INT(RND*16)
372:ARR=ARR*16:PRINT:NEXT I
373:FOR I=0 TO 15:ARR=INT(RND*16)
374:ARR=ARR*16:PRINT:NEXT I
375:FOR I=0 TO 15:ARR=INT(RND*16)
376:ARR=ARR*16:PRINT:NEXT I
377:FOR I=0 TO 15:ARR=INT(RND*16)
378:ARR=ARR*16:PRINT:NEXT I
379:FOR I=0 TO 15:ARR=INT(RND*16)
380:ARR=ARR*16:PRINT:NEXT I
381:FOR I=0 TO 15:ARR=INT(RND*16)
382:ARR=ARR*16:PRINT:NEXT I
383:FOR I=0 TO 15:ARR=INT(RND*16)
384:ARR=ARR*16:PRINT:NEXT I
385:FOR I=0 TO 15:ARR=INT(RND*16)
386:ARR=ARR*16:PRINT:NEXT I
387:FOR I=0 TO 15:ARR=INT(RND*16)
388:ARR=ARR*16:PRINT:NEXT I
389:FOR I=0 TO 15:ARR=INT(RND*16)
390:ARR=ARR*16:PRINT:NEXT I
391:FOR I=0 TO 15:ARR=INT(RND*16)
392:ARR=ARR*16:PRINT:NEXT I
393:FOR I=0 TO 15:ARR=INT(RND*16)
394:ARR=ARR*16:PRINT:NEXT I
395:FOR I=0 TO 15:ARR=INT(RND*16)
396:ARR=ARR*16:PRINT:NEXT I
397:FOR I=0 TO 15:ARR=INT(RND*16)
398:ARR=ARR*16:PRINT:NEXT I
399:FOR I=0 TO 15:ARR=INT(RND*16)
400:ARR=ARR*16:PRINT:NEXT I
401:FOR I=0 TO 15:ARR=INT(RND*16)
402:ARR=ARR*16:PRINT:NEXT I
403:FOR I=0 TO 15:ARR=INT(RND*16)
404:ARR=ARR*16:PRINT:NEXT I
405:FOR I=0 TO 15:ARR=INT(RND*16)
406:ARR=ARR*16:PRINT:NEXT I
407:FOR I=0 TO 15:ARR=INT(RND*16)
408:ARR=ARR*16:PRINT:NEXT I
409:FOR I=0 TO 15:ARR=INT(RND*16)
410:ARR=ARR*16:PRINT:NEXT I
411:FOR I=0 TO 15:ARR=INT(RND*16)
412:ARR=ARR*16:PRINT:NEXT I
413:FOR I=0 TO 15:ARR=INT(RND*16)
414:ARR=ARR*16:PRINT:NEXT I
415:FOR I=0 TO 15:ARR=INT(RND*16)
416:ARR=ARR*16:PRINT:NEXT I
417:FOR I=0 TO 15:ARR=INT(RND*16)
418:ARR=ARR*16:PRINT:NEXT I
419:FOR I=0 TO 15:ARR=INT(RND*16)
420:ARR=ARR*16:PRINT:NEXT I
421:FOR I=0 TO 15:ARR=INT(RND*16)
422:ARR=ARR*16:PRINT:NEXT I
423:FOR I=0 TO 15:ARR=INT(RND*16)
424:ARR=ARR*16:PRINT:NEXT I
425:FOR I=0 TO 15:ARR=INT(RND*16)
426:ARR=ARR*16:PRINT:NEXT I
427:FOR I=0 TO 15:ARR=INT(RND*16)
428:ARR=ARR*16:PRINT:NEXT I
429:FOR I=0 TO 15:ARR=INT(RND*16)
430:ARR=ARR*16:PRINT:NEXT I
431:FOR I=0 TO 15:ARR=INT(RND*16)
432:ARR=ARR*16:PRINT:NEXT I
433:FOR I=0 TO 15:ARR=INT(RND*16)
434:ARR=ARR*16:PRINT:NEXT I
435:FOR I=0 TO 15:ARR=INT(RND*16)
436:ARR=ARR*16:PRINT:NEXT I
437:FOR I=0 TO 15:ARR=INT(RND*16)
438:ARR=ARR*16:PRINT:NEXT I
439:FOR I=0 TO 15:ARR=INT(RND*16)
440:ARR=ARR*16:PRINT:NEXT I
441:FOR I=0 TO 15:ARR=INT(RND*16)
442:ARR=ARR*16:PRINT:NEXT I
443:FOR I=0 TO 15:ARR=INT(RND*16)
444:ARR=ARR*16:PRINT:NEXT I
445:FOR I=0 TO 15:ARR=INT(RND*16)
446:ARR=ARR*16:PRINT:NEXT I
447:FOR I=0 TO 15:ARR=INT(RND*16)
448:ARR=ARR*16:PRINT:NEXT I
449:FOR I=0 TO 15:ARR=INT(RND*16)
450:ARR=ARR*16:PRINT:NEXT I
451:FOR I=0 TO 15:ARR=INT(RND*16)
452:ARR=ARR*16:PRINT:NEXT I
453:FOR I=0 TO 15:ARR=INT(RND*16)
454:ARR=ARR*16:PRINT:NEXT I
455:FOR I=0 TO 15:ARR=INT(RND*16)
456:ARR=ARR*16:PRINT:NEXT I
457:FOR I=0 TO 15:ARR=INT(RND*16)
458:ARR=ARR*16:PRINT:NEXT I
459:FOR I=0 TO 15:ARR=INT(RND*16)
460:ARR=ARR*16:PRINT:NEXT I
461:FOR I=0 TO 15:ARR=INT(RND*16)
462:ARR=ARR*16:PRINT:NEXT I
463:FOR I=0 TO 15:ARR=INT(RND*16)
464:ARR=ARR*16:PRINT:NEXT I
465:FOR I=0 TO 15:ARR=INT(RND*16)
466:ARR=ARR*16:PRINT:NEXT I
467:FOR I=0 TO 15:ARR=INT(RND*16)
468:ARR=ARR*16:PRINT:NEXT I
469:FOR I=0 TO 15:ARR=INT(RND*16)
470:ARR=ARR*16:PRINT:NEXT I
471:FOR I=0 TO 15:ARR=INT(RND*16)
472:ARR=ARR*16:PRINT:NEXT I
473:FOR I=0 TO 15:ARR=INT(RND*16)
474:ARR=ARR*16:PRINT:NEXT I
475:FOR I=0 TO 15:ARR=INT(RND*16)
476:ARR=ARR*16:PRINT:NEXT I
477:FOR I=0 TO 15:ARR=INT(RND*16)
478:ARR=ARR*16:PRINT:NEXT I
479:FOR I=0 TO 15:ARR=INT(RND*16)
480:ARR=ARR*16:PRINT:NEXT I
481:FOR I=0 TO 15:ARR=INT(RND*16)
482:ARR=ARR*16:PRINT:NEXT I
483:FOR I=0 TO 15:ARR=INT(RND*16)
484:ARR=ARR*16:PRINT:NEXT I
485:FOR I=0 TO 15:ARR=INT(RND*16)
486:ARR=ARR*16:PRINT:NEXT I
487:FOR I=0 TO 15:ARR=INT(RND*16)
488:ARR=ARR*16:PRINT:NEXT I
489:FOR I=0 TO 15:ARR=INT(RND*16)
490:ARR=ARR*16:PRINT:NEXT I
491:FOR I=0 TO 15:ARR=INT(RND*16)
492:ARR=ARR*16:PRINT:NEXT I
493:FOR I=0 TO 15:ARR=INT(RND*16)
494:ARR=ARR*16:PRINT:NEXT I
495:FOR I=0 TO 15:ARR=INT(RND*16)
496:ARR=ARR*16:PRINT:NEXT I
497:FOR I=0 TO 15:ARR=INT(RND*16)
498:ARR=ARR*16:PRINT:NEXT I
499:FOR I=0 TO 15:ARR=INT(RND*16)
500:ARR=ARR*16:PRINT:NEXT I
501:FOR I=0 TO 15:ARR=INT(RND*16)
502:ARR=ARR*16:PRINT:NEXT I
503:FOR I=0 TO 15:ARR=INT(RND*16)
504:ARR=ARR*16:PRINT:NEXT I
505:FOR I=0 TO 15:ARR=INT(RND*16)
506:ARR=ARR*16:PRINT:NEXT I
507:FOR I=0 TO 15:ARR=INT(RND*16)
508:ARR=ARR*16:PRINT:NEXT I
509:FOR I=0 TO 15:ARR=INT(RND*16)
510:ARR=ARR*16:PRINT:NEXT I
511:FOR I=0 TO 15:ARR=INT(RND*16)
512:ARR=ARR*16:PRINT:NEXT I
513:FOR I=0 TO 15:ARR=INT(RND*16)
514:ARR=ARR*16:PRINT:NEXT I
515:FOR I=0 TO 15:ARR=INT(RND*16)
516:ARR=ARR*16:PRINT:NEXT I
517:FOR I=0 TO 15:ARR=INT(RND*16)
518:ARR=ARR*16:PRINT:NEXT I
519:FOR I=0 TO 15:ARR=INT(RND*16)
520:ARR=ARR*16:PRINT:NEXT I
521:FOR I=0 TO 15:ARR=INT(RND*16)
522:ARR=ARR*16:PRINT:NEXT I
523:FOR I=0 TO 15:ARR=INT(RND*16)
524:ARR=ARR*16:PRINT:NEXT I
525:FOR I=0 TO 15:ARR=INT(RND*16)
526:ARR=ARR*16:PRINT:NEXT I
527:
```





## Musical. Un buen paquete de programas musicales para el C-64

Una de las principales características del C-64 es su capacidad de sintetizar sonidos. De ello se encarga el SID (Sound Interface Device), un chip que controla todas estas funciones.

No obstante, la programación de música en el C-64 no es nada fácil. No existen comandos en BASIC para ello y el método de los POKES resulta complicado y aburrido —además de difícil de entender—.

No es extraño pues que las casas de software hayan diseñado programas musicales, desde los más sencillos hasta los más complicados, e incluso algunos como teclados de órgano para conectarlos directamente al ordenador.

### Características generales

El hardware necesario para utilizar este paquete de programa es un Commodore-64 y una unidad de disco. La impresora es opcional, pero conviene tenerla si se usa el MUSICALC 2.

El paquete de programas, diseñado por WAVERFORM y comercializado en España por IBM ALUIC, está compuesto por tres MUSICALs, el Sintetizador y Secuenciador, el "Sequencer", teclado de partituras, y el "Keyboard maker" configurador del teclado. También se incluyen el "Musical african/latin rhythm template" y el "Musical new wave & rock template" que contienen acompañamientos y melodías preprogramadas.



### MusicalC 1

Es el programa principal. Controla la música y controla la forma de las ondas, los filtros, voces, etc. Se trata de un programa libreto, es decir, medio básico medio código máquina.

La parte más característica del MUSICALC es el PANEL DE CONTROL. Cuenta de las siguientes partes:

A la derecha se encuentra la "repita", es la que aparecen las voces (con distintos volúmenes) mezclándose de



quien a la izquierda está preprogramada compuesta de 15 ondas de 16 columnas cada una. El total de ondas programables, teniendo en cuenta que las ondas y las melodías diferentes de que se dispone pueden encadenarse, es de 7.680 ondas.

En la parte superior izquierda están los controles de las tres voces, independientes entre sí. Cada una de ellas puede variar el tipo de onda y el tipo de envolvente (ataque, decaimiento, sostenimiento, relajación).

En la parte inferior derecha se encuentran los interruptores para conectar/desconectar los filtros, el tipo de filtro que se utiliza (pasabajas, paso banda, paso alto, decaimiento, etc.), y los tres filtros y envolventes debajo de ellos. Los controles temporizadores, para variar la velocidad de la música.

En la parte inferior derecha hay dos indicadores, "sound" y "voice". El primero de ellos indica el tipo de sonido, es decir la posición de los controles de la parte izquierda de la pantalla. El segundo es el indicador del número de melodía que se está ejecutando en ese momento.

Todos los controles de la parte izquierda de la pantalla son del tipo "mando deslizante", es decir, que se pueden mover arriba y abajo. Para regularlos hay que usar las teclas de flecha.

Además de estos controles deslizantes están los "interruptores" —se usan para seleccionar el tipo de onda, por ejemplo—

y se conectan y desconectan del mismo modo.

El ajuste a cada uno de estos tipos de controles se realiza pulsando primero la tecla "flecha hacia arriba" para entrar en el panel de control y a continuación pulsando una de las teclas de la parte izquierda del ordenador. Las teclas se corresponden con la posición que ocupan los controles en la pantalla.

La elección entre mandos deslizantes o interruptores se realiza con las teclas del cursor.

Para realizar operaciones más avanzadas se tiene acceso desde el panel de control a cargar una melodía, por ejemplo —hay que pulsar simultáneamente las teclas SHIFT y RI LL RH, con lo que nos encontramos ante el menú de opciones—.

Este menú ofrece muchas posibilidades, como borrar canciones, iniciar o detener el disco, etc.

### El manual de instrucciones

El único programa que se suministra, con manual es el MUSICALC 1.100 y el tres carpetas de él, pues es imprescindible en cualquier momento, poder acceder al ordenador sin necesidad de responder con una información correcta y precisa. Los dos discos de este paquete, el "New wave & rock template" y el "African/latin rhythm template" tampoco lo llevan, con lo que conviene tenerlos como datos de melodías preprogramadas.

Todo el manual, como los textos que aparecen en la pantalla del ordenador están en inglés, aunque sabemos que existe una versión en castellano y se los programas también traducidos que queda que comercialice IBM ALUIC.

Las veinte y dos páginas que componen el manual comienzan hablando de lo que es el MUSICALC 1, de la gramática de las premisas que se han de tomar al usar los programas en el disco, de la forma de hacer funcionar el programa.

A continuación viene un capítulo del panel de control con indicaciones sobre cada una de sus partes. También habla sobre los conceptos básicos para programar música y de responsabilidad del programa.

Los realmente sencillos aprendiendo a programar música, pues el manual indica paso a paso las teclas que hay que pulsar y luego pregunta y responde al ordenador.

También contiene un vocabulario de características, una especie de glosario palabra por palabra.

En su versión de la parte más completa del manual, explica qué es y para qué sirve el comando/función/tecla/pulsador, cómo se puede acceder desde el teclado, la barra, el panel o, explicaciones sobre lo que es la forma de una onda, los programas, extensiones que lleva dentro el DOS, los tipos de libro y las palabras clave.

El libro que aparece son unas notas sobre los controles más importantes, un índice resumen sobre los comandos del panel de control, del menú y sobre los programas internos.

### Musical2, "Scorewriter"

Este programa viene sin manual de instrucciones, pero la verdad es que no es necesario. Sirve para escribir partituras, se vea en la pantalla o en la impresora.



Con la impresora, o con el algún problema. El programa está diseñado para funcionar solo con la COMMODORE 128 o con una EPSON con interfaz C-ARXID o con cualquier otro nombre, ninguna de las dos, no hemos conseguido sacar ninguna partitura a impresora, pero tenemos entendido que la calidad gráfica es excelente.

Para hacer aparecer la partitura por la pantalla, hay que apagar primero unos valores, tales como el número de notas que van a aparecer en cada línea, el número de "páginas" que se van a imprimir, etc.

Este programa necesita al MUSICAL1 para funcionar, por lo que resulta difícil por sí solo.

### Musical3, el "Keyboard maker"

Serve para crear configuraciones musicales en el teclado de acorde con las necesidades de cada uno. Viene acompañado de unos 40 teclados prepro-

gramados", es decir configuraciones de teclado que van desde la música clásica hasta el rock.

Este programa también necesita el MUSICAL1 para funcionar.

Los dos discos de acompañamiento, tanto el de rock como el de ritmos latinos y africanos, dependen completamente del MUSICAL1. Contienen canciones pequeñas y sonidos que pueden ser alterados desde el panel del MUSICAL1 para lograr nuevos efectos, al igual que con los programas de demostración que vienen con el MUSICAL1.

### Conclusiones

Waveform ha creado un conjunto de programas que aparecen perfectamente las posibilidades del 64. Los tres son de una calidad excepcional, especialmente el primero. Es de suponer que Waveform continuará con la "suave" de los MUSICAL2, tanto con nuevos programas como con discos de acompañamiento.

El manual es ejemplar. Todos los detalles de los programas aparecen claramente explicados y con un buen número de ejemplos.

El único defecto que le hemos encontrado es que no puede usarse con la impresora MP800, tan popular entre los usuarios de Commodore.

Por último decir que, si bien este paquete de programas podría ser usado por cualquiera, está más bien dirigido a la gente que tiene algunos conocimientos de música o a los estudiantes o a los profesionales.

Este paquete musical ha sido diseñado y programado por Richard Wadman, Bill Modison, John Shrigard, Michael Miller y Chris Glegg. La música de demostración y acompañamientos fue creada por Tom Wesel y Harry Likas.

Todos estos programas los comercializa en España IDEALOGIC.

El precio del paquete completo es de 28.550 ptas.

También los venden sueltos al precio de 9.899 el MUSICAL1 y 7.105 el MUSICAL2 y MUSICAL3.

Los DEMOFILES van incluidos en el paquete completo, pero por separado valen 5.467 ptas. cada uno.

Para los que queráis ponerlo en contacto con IDEALOGIC, su dirección es IDEALOGIC, S.A.

c/ Gran Vía de Carlos III, 97 K.

08028 Barcelona

Y su teléfono al (93) 330 33 00

## Música para todos

Supongo que si todo el mundo es un poco tonto, ya debería que en algún momento se habrá imaginado subido a un escenario tocando un instrumento de teclado con una solista y ante de desprecupado que digan asombrado al público. De verdad, siempre ha quedado un poco sospechoso la facilidad con que el teclado de ritmo entre sonidos, ritmos, acompañamientos, etc., mientras que la estrella de ritmo le presta poca atención y, eso sí, siempre



COMPUTERS, S.A.

PAMPLONA, C. - Oficio al Realizado, la ciudad - Tel. 27 64 64 - Código Postal 31007  
34N.32.845.1145. Plaza de Bilbao, 1 - Tel. 41 62 37 - Telex: 30095-1477 - Cód. Post. 28005

## ¡¡PRECIOS ESPECIALES PARA COMERCIANTES!!

COMMODORE - 64  
COMMODORE - 16  
ZX SPECTRUM - 48K  
SPECTRUM PLUS  
QL SINCLAIR  
AMSTRAD  
MSX - GOLDSTAR

PERIFERICOS, PROGRAMAS, LIBROS NACIONALES, EXTRANJEROS, ETC.

6 MESES DE GARANTIA PARA ORDENADORES Y PERIFERICOS



La Generalitat de Catalunya Organiza el 1º Festival de Software



# Hoy, en solitario.

Dentro de muy pocos años, la informática será un instrumento imprescindible para el conocimiento y el desarrollo de la sociedad. Y para construir este futuro, es necesario que hoy surjan, se promocionen y se ponga de relieve la labor silenciosa y en solitario de muchos nuevos valores que con toda seguridad existen.

Para hacerlo posible, la Generalitat de Catalunya organiza el 1º Festival de Software. A él pueden enviar sus trabajos todas aquellas personas o grupos que hayan creado programas informáticos.

La fecha límite para la presentación de programas, será el 30 de Mayo, y en Octubre se exhibirá públicamente los seleccionados, que serán puntuados por el público, con cuyos criterios y los del jurado se entregarán los diversos premios, que incluyen una categoría juvenil.

Participa. Y ojalá éste sea tu primer paso hacia el éxito.





# Mañana, el éxito.

## Deseo toda la información

Deseo recibir las bases para participar en el 1<sup>er</sup> Festival de Software

Tipo de programas que desearía presentar. Programas en ☐ catalán ☐ castellano

Ordenador necesario (marca y modelo) \_\_\_\_\_

Pantalla ☐ Color ☐ B/N Impresora (si es necesaria) \_\_\_\_\_

Otros periféricos o extensiones \_\_\_\_\_

Nombre y apellidos \_\_\_\_\_ Edad \_\_\_\_\_

Dirección \_\_\_\_\_ Ciudad \_\_\_\_\_

Envíe este cupón al Centre Divulgador de la Informàtica. Ap. de Correos 5185 Barcelona.



**GENERALITAT  
DE CATALUNYA**  
**CENTRE  
DIVULGADOR DE  
LA INFORMÀTICA**

EL CENTRE D'AJUDA TÈCNICA  
DE LA INFORMÀTICA  
ORGANITZADOR DEL FESTIVAL  
DE SOFTWARE EN COL·LABORACIÓ  
AMB LA ASSOCIACIÓ DE  
TÈCNICS DE L'INFORMÀTICA I  
INFORMÀTICA DE BARCELONA

con un solo dedo. Se imagina que una máquina así de completa supone un desembolso muy fuerte por lo que sorprende un poco saber que todo esto lo tienes disponible en tu C-64.

A continuación vamos a comentar unos programas que nos han dejado realmente impresionados y que merecen una vez más el respeto que le tenemos a nuestro querido C-64. Estos programas son instructivos y divertidos a la vez, para el estudiante de música o para el principiante que siempre ha querido y nunca ha podido. Son, simplemente, tan complicados o tan sencillos como las necesidades del usuario. Mas que unos programas de música, son toda una experiencia. Quien sabe, pueden ayudarte a descubrir un talento hasta este momento desconocido, dando rienda suelta a tu creatividad.

### Kawasaki Rhythm Rocker

Este programa (que realmente son dos en uno: uno de música y uno de gráficos) es una combinación impresionante de ritmo, efectos de percusión y batería, sonidos sintetizados, y efectos gráficos, disponibles al pulsar un botón para que des rienda suelta a tu imaginación.

El programa de música te permite:

1. Tocar el sintetizador con una serie de acompañamientos pregrabados de bajo.
2. Tocar el sintetizador con un acompañamiento de bajo y ritmo de tu propia creación.
3. Tocar efectos de batería y percusión con el bajo.
4. Grabar tus propias composiciones.
5. Tocar números de bajo y percusión creados con otro programa de Kawasaki "The Composer".

El programa de gráficos te permite:

1. Generar dibujos gráficos multicolores incorporados en el software.
2. Modificar el tamaño, forma y ángulo de los gráficos.
3. Paralelizar el dibujo en cualquier momento para hacer un scroll en pantalla.



4. Añadir otros efectos especiales de colores y tamaño a los gráficos.

Las dos filas superiores de teclas constituyen el teclado tipo piano, que quedan disponibles para tocar al cargar el programa. En este momento estamos en el modo sintetizador, y pulsando la tecla K podemos pasar a un efecto que "dobla" la nota, produciendo un curioso efecto gráfico a la vez y pulsamos la tecla de escape. Pulsando la tecla Shift se produce el vibrato, y la Shift Lock, para sostenimiento y vibrato.

La tecla Return nos remite a una pantalla de ayuda donde se encuentra un resumen de todas las modalidades. Se vuelve a la pantalla principal pulsando la tecla S. Las tres modalidades son: sostenido (SUS), bajo (BASS) y Percusión (PERCS). La tecla F7 te permite cambiar de modo y un cuadrado pequeño a la derecha de la pantalla te indica el modo en el que te encuentras en cualquier momento.

En el modo de percusión el teclado desaparece y lo sustituye una "reja" en pantalla. En este modo cada tecla musical produce una percusión o un efecto sonoro diferente. Las teclas 3, 9, 0, —, HOME, y \* abren la puerta a unos curtos visuales especiales que aparecen en pantalla anunciando su salida con unos sonidos de cornos galanos.

Pulsando la tecla 13 dos veces pasamos al modo del bajo que te permite tocar el bajo acompañado por el ritmo de fondo. Este modo (y el de sintetizador) también puede llevar los gráficos de "reja" si lo deseas. El programa lleva incorporado cuatro acompañamientos de bajo que puedes seleccionar usando la tecla F1, pero también puedes cargar otros acompañamientos de disco.

En la parte gráfica del programa todas las combinaciones están bien explicadas en el manual (disponible en castellano), pero hacemos un pequeño resumen aquí. La tecla X sirve para empezar a dibujar y la tecla Z, para salir. La tecla D modifica el ángulo de la línea que estás dibujando, y la tecla B para elegir uno de cuatro tamaños. La tecla G varía la anchura de los dibujos y la tecla H borra cualquier dibujo previo y empieza uno nuevo que comienza durante un periodo de tiempo más largo. Este representa unas líneas más anchas y más largas. El propio programa también crea dibujos al azar y cuando termine uno empieza otro nuevo. Si practicas un poco, sabrás el momento más oportuno para parar un dibujo para que acompañe perfectamente el sonido, creando un número musical y gráfico usando tu propia imaginación. Cada vez que pulsas la tecla C, aparece una de las dieciséis combinaciones de color. La tecla M hace un scroll con el dibujo entero, la N para el scroll, la A te devuelve a la pantalla principal.

Entrar otra pantalla gráfica pulsando

CTRL y Return en el modo del teclado que se llama Gráfica de Sonido. El sonido se representa de una forma gráfica mediante unos cuadrados y rombos que aparecen de acuerdo con las notas tocadas y con el ritmo de la pista.

El "Dark Lounge" (Salón del Oso) es una modalidad que te permite trabajar directamente con la unidad de discos. Las funciones y las teclas que las controlan son las siguientes:

F1 — Para veraciones simplificadas de los comandos más utilizados (FORMAT, SEARCH, RENAME, COPY, INITIALIZE, VALIDATE).

F2 — Para salvar tus composiciones o secuencias gráficas en disco.

F3 — Para acceder al directorio del disco.

F7 — Para cargar un fichero almacenado en el disco "Rhythm Rocker" o cualquier otro tipo.

Puedes grabar en la memoria el ordenador (tecla —) y volver a escuchar desde la tecla a la izquierda una melodía, o un acompañamiento pasando cualquiera de las dos funciones con la tecla C, comandos que pueden grabar hasta 256 notas. El color del borde de la pantalla cambia a rojo al llegar a 250 notas. Es importante observar esto porque, si grabas más de 256 notas, se borrará como una nueva grabación y se borrarán todas las notas anteriores. Las teclas de cursor te permiten controlar la velocidad de la melodía o acompañamiento.

Existen dos funciones de dibujo para tocar y grabar el sintetizador o la percusión encima del bajo. Pulsando la tecla 13 se selecciona DUB-1 (para el sintetizador) o DUB-2 (para sonidos y ritmos de percusión).

Este programa es un paquete musical y gráfico realmente completo, pensando la claridad de las instrucciones en el manual, no esperar ningún resultado, incluso para una persona que no se pueda de música ni de programación, proporciona un medio de creación de líneas para aquellos que ya se distraen de unos acontecimientos de estos terrenos.

### Kawasaki Synthesizer

Este paquete incluye dos discos "The Performer" (El Interpretador) y "The Composer" (El Compositor). El que vamos a comentar primero es el más sencillo de manejar: el Interpretador.

Desde el este programa tendrás que ver opciones que vamos a explicar a continuación. Todas están detalladas en la pantalla de ayuda (pulsando la tecla H).

1. Como tocar el teclado. Una serie de instrucciones en pantalla que repiten la disposición de las teclas.

2. Escuchar melodías, creadas por Ryō Kawajike. Existen 11 canciones.





grabados, acompañados por una representación gráfica.

3. El uso del programa de demostración que te presenta a otros personajes famosos.

4. Una opción te permite seleccionar uno de 21 sonidos grabados.

5. Como modificar el rango de las notas del teclado o tocar en modo polifónico.

6. Como modificar las formas de onda.

7. 2 activa la triangular (piano, flauta, 14 activa la diatónica de serra (trompeta, violín), 16 activa la onda pulsante (guitarra, clarinete), 18 activa el ruido blanco (trueno, viento, ondas).

7. El uso para tocar la introducción y la canción del Kawasaki Synthesizer.

Haciendo uso de todas las combinaciones posibles puedes llegar a producir hasta 300 sonidos diferentes.

Si "13 Interprete" sirve para introducirte en el mundo de los sonidos sintetizados de hecho, este programa es ideal como introducción a todos los programas de Ryo Kawasaki, "La Composición" te deja mucho más margen para experimentar y explorar todas las posibilidades que se nos ofrecen. Las siguientes funciones del programa te permiten:

1. Tocar música utilizando una antiprisma para de sonidos y efectos especiales.

2. Grabar y volver a escuchar el bajo para poder usarlo como acompañamiento al otro compás.

3. Grabar y reproducir composiciones de hasta tres voces. También disponemos de ejemplos.

4. Crear sonidos y efectos especiales y almacenarlos en disco.

5. La pantalla del teclado se permite jugar con los siguientes posibilidades:

11, 13, 15, 17: juego entre cuatro octavas alta, mediana, baja y superbaja.

12, 14, 16, 18: juego entre la onda triangular, onda diatónica de serra, onda pulsante, ruido blanco.

1. Activar efectos Wah-Wah.

2. Desactivar este efecto.

Barra de espacio: Pantalla para editar sonidos.

—>—>: Cargar el sonido.

CHSR: V. Monofónico.

CHSR: A la derecha: Polifónico.

Tecla de dos puntos: 1. Activa el vibrato y el punto y coma: 2. lo desactiva.

La pantalla de editar sonidos te permite realizar una serie de funciones importantes:

1. Editar sonidos individuales.

2. Salvar y cargar sonidos grabados en disco.

3. Grabar, editar, y reproducir secuencias (composiciones).

4. Salvar y cargar las secuencias en disco.

Desde la pantalla de editar sonidos se puede acceder al Modo de Mezclar pulsando la tecla M. Esta modalidad te permite elegir una forma de onda diferente para cada uno de las tres voces.

Hemos resumido brevemente las funciones principales del "Kawasaki Synthesizer" para dar una idea de la versatilidad de estos programas. Las instrucciones del manual (traducido al castellano) son fáciles de seguir, lo que te permite ponerle a tocar, jugar, experimentar, componer, aprender y en general, pasarlo muy bien nada más cargado el programa.

### 3001 Sound Odyssey

En mi opinión nos encontramos frente a uno de los mejores programas musicales del mercado, tratase de reseñar las cualidades que me han llevado a esta conclusión.

El programa viene acompañado de un manual detallado y correctamente traducido al castellano. El nivel técnico del manejo del programa y el descubrimiento de sus posibilidades.

Al cargar el programa nos encontramos frente a la presentación hasta que pulsamos una tecla, en ese momento pasaremos al menú principal, en el que podremos ver las opciones de sonido, el tutorial sobre síntesis de sonido, la demostración y la opción de la presentación. Para seleccionar la opción podemos utilizar el joystick o el teclado (tanto en esta como en todas las demás opciones del programa).

Si para empezar vemos la demostración, nos dejara con la boca abierta y nos dara ánimos para comenzar el aprendizaje del manejo del programa. En la demostración visualiza en pantalla los ajustes que realiza sobre los distintos mundos del sintetizador mientras escuchamos los efectos producidos sobre el sonido.

Posteriormente podemos pasar al tutorial, en el podemos aprender todo aquello que necesitamos saber sobre la síntesis de sonido. En el resto del tutorial encontramos una lista de nueve secciones, en cada una de ellas tendremos tres opciones: una informativa (Show me), consistente en explicaciones breves, otra didáctica con demostración visual y sonora y la tercera tomar el control

sobre esa parte del sintetizador para experimentar por nosotros mismos (En me).

Los temas estudiados son: introducción, volumen, pitch, ADSR, formas de onda, filtros, modulación en amplitud y sincronización, modulación con LFO, sintetizador de envolvente.

El único problema es que se trata apenas de ejemplos, pero los ejemplos que incorpora son tantos y tan claros que en realidad no importa mucho. Quisiera si una distribuidora lo tradujera al castellano, pero no parece tener intención de hacerlo, aunque el manual ya está traducido.

En otras opciones que nos quedan son la de la presentación, que la hemos visto al cargar el programa, y la más importante, que es la utilización práctica del sintetizador. Cuando seleccionamos esta opción pasamos a cinco pantallas frente a: teclado de un sintetizador con un teclado negro para de cuatro octavas completas en el que vemos la nota pasada, los controles del sintetizador son:

Volumen, y para cada una de las voces interruptores, ADSR, forma de onda, estructura de impulsos, modulación en amplitud, sincronización, amplitud (pitch), filtros, control de onda, parabanda, parámetros, parabanda, control de sincronización, control, velocidad de LFO, sus filtros, seguidor de envolvente, etc.

Para escuchar una muestra disponemos de las dos filas superiores del teclado, la superior son las teclas negras y la inferior las blancas, distribuidas igual que en un piano. Y disponemos también de los controles de onda y de



podemos acceder a los controles de onda de un instrumento de onda y de onda. Para cambiar las octavas son, las que estamos tocando, pulsamos las teclas correspondientes (shift, arrow).

Además se puede acompañar esta ruta con cambios de color en la pantalla de la pantalla, para el control de onda y de onda. A continuación se muestra una muestra de la notación de la nota y de onda, se pueden pulsar la tecla y de onda. Los controles actúan a modo de interruptores ON/OFF.

Todos los mundos del sintetizador son

podemos regalar con la ayuda del joystick a las teclas de cursor y la de flecha a la izquierda, con este el sombrero se puede situar en el modo directo, mientras que estamos así metidos en el teclado, pero como la mayoría de los juegos que, posiblemente desearíamos obtener sean complicados, es necesario pasar al estado de las posibilidades de *DirectX* y *Secundario*.

Disponemos de un número de treinta y nueve prescotes y cinco cartas programables, además de los ocho instantáneos asignados a las tarjetas de baraja. El diseño de estos prescotes puede ser bastante motivador, situado en el plano cartográfico en el que, por lo que podemos trabajar para entregar la variedad de sonidos que desearíamos. Si nuestra intención es diseñar para influir en positivamente. Para obtener los nombres de los trabajos ya existentes en el disco prescotes a la opción LUDIC y teórica, nos DIR (clon) (en)

También disponemos de un sector a cada 100 ha que podemos mecanizar, es decir, seleccionar medidas desde el suelo. El sistema también tiene un control de riego R, con un margen de ajuste desde -99 hasta +99, lo que permite disminuir o aumentar, según las condiciones de humedad en el suelo, la cantidad de agua aplicada, desde 0 hasta 100% del riego normal, corrigiendo o con ayuda de otros sensores las posibles representaciones. El riego lo hacemos en la humedad que puede causar el crecimiento.

En el caso de programa podemos encontrar un par de ejemplos de intrínsecos: *matrices* (definido en *defmat*) y *demo21*, ambos de procedencia *iprevnet*, *bulgamer*, *canon*, *canon* y *bulgamer*. Los ejemplos se permiten demostrar un camino a la unidad *canon* *canon*, dejando la meta *palabra* y cambiando los *objetos* a *matrices* *demo21* *matrices* *iprevnet*.

Esos que es difícil encontrar los perfiles  
duras de este programa en una revista, y  
es mejor ver el disco de demostración  
y uno de los distribuidores.

### Music processor

... como programador creo que este es J.p. porque que mas me ha gustado, ya que dispone de un nuevo lenguaje de programación totalmente destinado a la música, con entintadores que permiten dibujar la letra a la música, marcar los ritmos, etc.

En la primera sección del manual se busca que, entre dos modos de aprendizaje, a través de un programa, el primero de ellos sea el más adecuado experimentalmente considerando un criterio, pasar el 7 para obtener un buen resumen de sus funciones y trabajar a priori todo, el segundo de ellos es más lento pero seguro, consiste en seguir paso a paso las secciones del manual sin controlarlo.



En el primer grupo de instrumentos, tenemos tres tipos de acorde de trébol que podemos subdividir. Para mostrar los ejemplos musicales, para cada sistema, elegiremos la tónica de  $a$  o  $A$ . Para pasar al modo de piano, pulsaremos la tecla  $B$ , y para retornarnos frente a un pentagrama doblado, con las claves de sol y fa, el terceros corda nota que pulsamos. Para subdividir instrumentos, entre los 99 disponibles, basta pulsar las teclas 1-9, para cambiar la octava sobre la que estamos, así como pulsamos 0-5 o 1-6, para cambiar la ordenación de los dedos. En super-modo, son las teclas de la octava más alta, las inferiores las de octava más baja. Si, queremos, activar el acompañamiento, pulsamos la tecla de flecha a la izquierda.

Se debe, en suma, preparar pruebas. El 1, estaremos escuchando una caja de música que interpretará todas las piezas del drama como dramas de otros.

En el E-2 pasamos al modo de grabación, toda la actividad se manifiesta conforme tocamos. Mas tarde podemos reproducir el grabado E-4 desde el menú principal. Si en el modo de grabación pulsamos se toca el RUTERRE aparece el cursor en el ángulo inferior izquierdo, ahora podemos tocar comandos de los que aparecen en el manual, por ejemplo podemos hacer que aparezca la otra muestra, sería la melodía

Con la ayuda de E-Fluencio, que se registra una vez al día, a las 18 pasamos al modo R-ANDM (aleatorio), en el que el computador toma la iniciativa y se dedica a producir sus propias composiciones. Este modo de funcionamiento es útil para indagar la variedad de sonidos que produce el instrumento al producirse con los teclas de cuerdas mientras el ordenador toma sus "obras maestras aleatorias".

Por último, con 15 parámetros al modo de edición (el más interesante para los programadores), es el modo de funcionamiento más potente del Mursi: Proceso, ya que podemos utilizar el editor como si estuviéramos trabajando con un programa en Basic, almacenando las líneas del 'programa' en la memoria

RAM, salvandolas con SAVE, cargandolas con LOAD.

Para abarcar líneas tendríamos un número de línea entre 1 y 1534 seguido de los comandos que desearíamos ejecutar. Para distribuir los comandos sobre el número de líneas y pulsamos **ILL** **IRN** (igual que en BASIC). Mientras se ejecuta una máquina podemos cambiar el mismo pulsando cualquiera de las teclas del zero al nueve. Hacemos pulsar, pasar al modo de **IRN** **CL** pulsando la tecla **L**, o al de paso a paso pulsando la **SOFT** **PI**, para volver a, o, incluso pulsar en la **P** (**PI**) **CL**. Si queremos detener la antepulsación de una máquina pulsamos la **Q** o **STOP**, y paramos al modo normal de edición de nuevo, reiniciando.

[illegible]

Comprenderlos que no tiene todas las posibilidades de este programa en todos los comandos que podemos utilizar, ya que tiene de todo tipo (para el mismo, ALDR, entre, etc.). La lista será larga, pero el manual es bastante claro y tiene más de sesenta páginas dedicadas a estos comandos de programación.

Lo más interesante es saber que disponemos de un lenguaje de programación musical con el que podemos hacer rítmico y una maravilla sin quebrarnos la cabeza con cantidad de pocos ramos y cosas de esas, ya que en comandos de control de la música se llaman por su nombre musical, por ejemplo "3 durato,





**TODOS LOS PROGRAMAS DE CADA LIBRO ESTAN A SU DISPOSICION EN DISCO O EN CASSETTE**

**FERRE - MORET S.A.**

TURIS, 8 ENTLO 2.º - ☎ 218 48 04 - 218 40 58  
TELEX 91681 CROCT E - 08006 - BARCELONA

**ESTAREMOS EN INFORMAT 85  
BARCELONA**

Con todo lo que se puede hacer con el Commodore 64, esta descripción detalladamente en este libro. Su lectura no es tan árida tan apasionante como la de una novela, sino que contiene, además de listas de útiles programas, sobre todo muchas, muchas aplicaciones realizables en el C64. Se ha valorado especialmente, que el libro sea de fácil comprensión por los no iniciados. Un estudio del sistema. El ordenador escribe poemas, arjetas de invitación, cartas publicitarias personalizadas, corre por kilómetro de su coche, cálculo de costes de construcción, calculadora de bolsillo, fichero de recetas, anuario, fichero personal de la salud, plus electrónico de dieta, diccionario inteligente, CAD para trabajos manuales, optimización de rutas, escape para publicistas, juegos de estrategia. En parte los listados de programas útiles para ser tecleados, aunque que ha sido posible condensar «recetas» en una o dos páginas. Si hasta el momento no sabe qué hacer con su Commodore después de leer este libro lo sabrá seguro!

**EL LIBRO DE IDEAS DEL COMMODORE 64, 1984, más de 200 páginas, ptas. 1.800,-**

**64 CONSEJOS Y TRUCOS**

**CONSEJOS Y TRUCOS**, con más de 75.000 ejemplares vendidos en Alemania, es uno de los libros más vendidos de DATA BOOKER. Es una colección muy interesante de ideas para la programación del Commodore 64, de POKEs y útiles rutinas e interesantes programas. Del contenido: Gráficos 3D en Basic - gráficos de barras en colores - definición de un juego propio de caracteres - simulación del zorro con el joystick - Basic para avanzados - el C64 habla castellano - CP/M en el Commodore 64 - conexión de impresoras a través del puerto de usuario - transmisión de datos desde y hacia otros ordenadores - aritmético en octario - recuperar un fichero, que no ha sido borrado correctamente - generar una línea Basic en Basic - el buffer del cassette como memoria de datos - multitasking en el Commodore 64 - la página cero - COTO, GOTO y RETURN con números de línea calculados - función INKEY y STREQ - repeticiones automáticas de todas las teclas. Todos los programas en lenguaje máquina con programas cargadores en Basic.

**64 CONSEJOS Y TRUCOS, 1984, 364 pág. P.V.P. 1.800,- ptas.**

**PEEKs & POKes PARA COMMODORE 64**

Con importantes comandos PEEK y POKE se pueden hacer también desde el Basic muchas cosas, para las que se necesitarían normalmente complejas rutinas en lenguaje máquina. Este libro explica de manera sencilla el manejo de PEEKs y POKes. Con una sencilla cantidad de POKes importantes y su posible aplicación. Para ello se explican perfectamente la estructura del Commodore 64. Sistema operativo, interpretador, página cero, apuntadores y stack, generador de caracteres, registros de sprites, programación de interfaces, descripción del cassette. Además una introducción al lenguaje máquina. Muchos programas ejemplo.

**177 pág. P.V.P. 1.600,- ptas.**

**MANUAL DEL CASSETTE PARA EL C64 Y VIC-20**

Un excelente libro, que le mostrará todas las posibilidades que le ofrece su grabadora de cassettes. Describe detalladamente, y de forma comprensible, todo sobre el cassette y la grabación en cassette. Con verdaderos programas para de venta: Accotest, Catalogo (busca y carga automáticamente), backup de y a disco, SAVE de areas de memoria, y lo más sorprendente: un nuevo sistema operativo de cassette con el 16-33 veces más rápido FastTape. Además otras subrutinas y programas de utilidad (ajuste de cabezales, algoritmo de control).

**190 pág. P.V.P. 1.600,- ptas.**

**MSX**

El libro contiene una amplia colección de importantes programas que abarcan, desde un desensamblador hasta un programa de clasificaciones deportivas, juegos superemocionantes y aplicaciones complejas. Los programas muestran además importantes consejos y trucos para la programación. Estos programas funcionan en todos los ordenadores MSX, así como en el SPECTROVIDEO 318/326.

**ESTRUCTURA DEL CONTENIDO**

- Válculo reverso hexadecimal. Editor gráfico. Editor de sonido.
- Estructura de ordenador. Luta silenciosa de variables.
- Calendario. Desensamblador.
- ADMINISTRACION de una colección de discos L.P.
- HOLLOW - JUEGO DE LAS CERRILLAS.
- DIAGRAMAS DE BARRAS.
- TABLAS DE POSITIVAS.

**194 pág. 1.985 P.V.P. 2.200,- ptas.**

**BOLETIN DE PEDIDO**

**FERRE - MORET S.A.**

Turis 8, 8.º entlo. 2.º Tel. 218 48 04  
BARCELONA 08006

Desde ahora

Gastos envío 300 ptas.

NOMBRE

DIRECCION

o Adjunto cheque o Remite más tarde el importe

var, velocidad, profundidad" activa el vibrato en la voz... con una velocidad... y con una profundidad de modulación. "Vibrato N" pone el vibrato en N, etc. Supongo que a muchos les agrada disponer de estos comandos para olvidarse de los pesados tanfidos de recordar.

Si encontramos que el teclado del Commodore resulta incómodo dado la disposición de las teclas, la misma cosa sucede con teclados tipo piano del tamaño del ordenador que ensaya encima de este creando un hueco para poder acceder a las teclas de función, de forma que puedes tocar tranquilamente sin confundirte. El programa que acompaña el teclado es una demostración muy completa de todos los programas de Ryu Kawasaki. Adicionalmente, este programa te permite tocar el teclado con el sonido de piano, sintetizador o bajo.

Artículo	Versión P.S.P.	
Inmutable musical		
Keyboard	Disk	13.700
Inmutable musical		
Keyboard	Cas	12.100
Musix processor	Disk	9.900
Musix processor	Cas	7.900
Kawasaki Synthtrecer	Disk	12.200
Kawasaki Synthtrecer	Cas	9.500
Kawasaki Synthtrecer	Disk	10.700
Kawasaki Synthtrecer	Cas	7.900
3000 Sound Objects	Disk	10.500
3000 Sound Objects	Cas	7.900

Estos programas son comercializados por la empresa Case de Software S.A.

c/ Aragón, 272, 8, 8  
08007 Barcelona  
Tel.: (93) 215 69 52

## Music Master

El Musix Master es un programa diseñado a facilitar el descubrimiento de las posibilidades del Commodore 64 para producir sonidos, permite variar los parámetros que controlan la forma de onda, envolvente y filtros que, una vez fijados para cada uno de las tres voces, hacen posible tocar una melodía desde el teclado como si se tratase de un piano. Por tanto, este programa funciona normalmente como un sintetizador programable monofónico, aunque con un poco de tiempo se pueden llegar a obtener acordes.

El Musix Master está dotado de un generador de ritmos automático que permite generar música de fondo o acompañamiento. Este generador de ritmos tiene un total de treinta y cuatro ritmos, y, si, usuario puede definir otros dos más.

Además de utilizarlo como si fuese un piano de acompañamiento, puede utilizarse en el modo programado. En este modo puede editarse una melodía nota a nota o al normal de ejecución, para



posteriormente, repetirla a la misma o diferente velocidad. Podemos programar y grabar hasta tres partituras una para cada personalidad, creando una melodía a tres voces.

Cuando editamos una canción disponemos de memoria para almacenar un máximo de mil quinientos notas para cada voz.

En la pantalla del ordenador podemos ver una sección del teclado de un piano representado o pentagramas, de modo que podamos algo normal para aquellos que no estén acostumbrados a la representación de notas en forma de letras. El manual dice que los músicos no tendrán dificultades para adaptarse a esta notación musical aunque nunca la he utilizado antes, y los programadores adaptarán pronto al pentagrama y a teclado de piano, pasando posiblemente al sistema de composición a tres voces con letras.

El Musix Master permite también crear efectos especiales con el chip de sonido del Commodore 64 (SID), entre ellos el wah-wah, vibrato, reverberación y tremolo, imitando el efecto oscilador para modular otros registros del SID. Los últimos efectos requieren algún tiempo para poder manipularlos, y no están dedicados a los principiantes, pero pueden intentarse cuando tengamos alguna praxis en el manejo de este programa.

Una característica más importante y peculiar del Musix Master es que incorpora un generador de programas Basic, que nos proporciona un modo de cubrir la falta de comandos de música en el Basic del 64.

Con este programa puedes generar una vibratura musical que podrás añadir a tus propios programas, para introducir solo líneas que editar la música simplemente utilizando los comandos normales del Musix Master, y emitir en la opción del generador de programas Basic, pulsando las teclas Shift y W. En este momento el programa le pregunta el número de línea nueva y el número entre líneas, le damos los valores que permitían colocar la numeración de líneas donde nos venga mejor en el rango de 0 a 65535, y luego hacer el SAVE.

Para utilizar la rutina musical que hemos generado, debemos hacer un MERGE con alguna de las extensiones Basic disponibles en el mercado (Simon's, Bridget's, VisiText, etc.) o alguna rutina de las apariciones en la sección de Magia. Una vez esto es cargado primero la rutina musical y luego teclear el resto del programa.

Debemos hacer una anotación, y es que los ritmos especiales y de acompañamiento no son pasados al programa generado en Basic, ya que todos estos materiales se suelen hacer un código máquina (podéis verlo en este mismo número en el artículo Servrote SID).

Como evaluación general, podemos decir que el programa está bastante bien, pero a medida que al nivel de posibilidades sonoras de un Kawasaki Synthtrecer o un Musix Processor, por otro lado la opción de poder generar melodías en Basic para incorporarlas a nuestros propios programas no la incluyen otros programas de mayor calidad musical. Hemos considerado que este es un programa para aquellos que quieren jugar en sus aspectos musicales de su ordenador sin gastar mucho dinero, para más adelante saber de más, con otro programa más completo en el caso de que le guste ese campo, aunque también lo podemos considerar como un utilitario para facilitar la edición de las rutinas musicales de nuestros programas.

Este programa lo comercializa en España la casa Sakate de Madrid a las ventas son:

Sakate  
Módulo, 14  
28028 Madrid  
Tel.: (91) 256 77 84

El precio del Musix Master en cinta es de 7.000 ptas. y la versión en disco 7.500 ptas.

## Music 64

El SIC 64 es un sintetizador monofónico polifónico que consiste en un teclado controlado por 64 (64) teclas, cada una dispone de tres atributos independientes que le permiten imitar una voz o sintetizador. Dispone de tres osciladores con forma de onda que van desde la triangular a la cuadrada, desde la seno, impulsos y ruido, la unidad de control ADSR, una amplitud fijo y fijo.





facilidad obvia de controlar los instrumentos en amplitud y frecuencia. Todo esto era difícilmente accesible para una persona que no fuera un programador muy experto, pero en este paquete también disponemos de un teclado.

El instrumento es bastante fácil de hacer funcionar porque basta con conectar el teclado a cable suministrado, y este último al sintetizador través del Commodore 14, programa ya en disco o cassette.

Eventualmente el uso del MUSE 64 se da en dos ámbitos diferentes: la definición de los instrumentos y la ejecución de música en tiempo real. En la definición de instrumentos, el sistema presenta una pantalla que simula el panel de un sintetizador, con la posibilidad de cambiar todos los parámetros visualizados. En estos casos, instrumentos potenciales en las 64 maneras de funcionamiento, pero como representantes de la facilidad de probar los resultados inmediatamente, este instrumento es idóneo para experimentar con todos aquellos efectos que se pueden hacer en la música hoy en día. En el modo de ejecución, basta pulsar una tecla de ordenador para que se reproduzca en seguida lo que nosotros hemos tan soñado. Si en este momento queremos cambiar de instrumento, basta con pulsar la tecla o teclas para volver a la situación anterior, preparados para crear nuevas melodías. Elgado a este momento, tenemos que plantearnos una pregunta: ¿Por qué, entonces, del 64, ¿cómo se la conectamos a los 64, ¿cómo se la conectamos a los 64 en el modo polifónico, se pueden crear melodías de donde han salido otras voces? Me gustaría de la programación.

El tiempo necesario se compone del teclado polifónico, unidad de disco o cassette, interfaz, y una de ellas es el programa "Organos Rítmicos".

El programa funciona en dos modos: 1. Modo de edición y 2. Modo de ejecución. En el modo de edición, se puede crear o modificar los ritmos y se puede hacer grabar los ritmos en el modo de ejecución, permitiendo la utilización de los 64 canales de la posibilidad de modificar la selección del ritmo o de los ritmos en tiempo real.

En el modo de edición, los ritmos se

ponen de las siguientes funciones:

—Crear un archivo de 9 ritmos diferentes.

—Modificar los ritmos.

—Dar un nombre a cualquier ritmo creado.

—Borrar un ritmo en disco.

—Cargar un ritmo de disco.

—El cambiar los ritmos, mientras se realizan las modificaciones.

Como crear o modificar un ritmo. El dicho ritmo o ritmo es una extensión de 128 bytes y ritmos de 7 instrumentos.

Los siguientes parámetros sonoros también se pueden modificar:

1. Velocidad
2. Resonancia
3. Ataque
4. Decaimiento
5. Sostenimiento
6. Relajación
7. Volumen
8. Transposición
9. Escala de onda

El precio del teclado y el software es de \$1.900. La empresa Alameda nos garantiza que la casa de ritmos está recién llegada y el precio de comercialización andará por las 14.000 pesetas.

La casa que comercializaremos posteriormente es:

**ALBAREDA**  
c/ Comer, 19  
Torreje  
Lleida

Tel. 1973 31 04 02, 31 23 51

## Commodore 128

Propone mucho, mucho más. El sistema de diseño de Commodore 128 dispone de todo lo que la casa Commodore ha avanzado, nos aporta un artículo de autor, un libro. De momento lo único que podemos hacer es enseñar a la casa Commodore a personalizar un par de datos más.



—128K, con ampliación hasta 512K.

—El potente lenguaje de programación que permite el uso de más de 100 sentencias y comandos.

—Pantalla de 25 líneas de 80 columnas y 25440 para compatibilizarlo con el C-64.

—Microprocesador 8502 y 2804MHz.

—El teclado principal tiene la misma disposición que el del C-64, además de un teclado numérico.

—Sistema operativo (P.M.S. Kern).

—Quizás su punto más fuerte será su compatibilidad con el C-64 y sus periféricos. ¿A ver si es verdad?

## El PC de Commodore



La segunda pregunta por la que se conoce es la presentación del modo de trabajo de la gama profesional de Commodore, decir que es totalmente compatible con el PC de la casa IBM, según es común y corriente, pronto será un hecho que hará temblar muchas marcas de ordenadores.

De momento solo sabemos el precio de lanzamiento, unos mil quinientos dólares, y tras el lanzamiento en Alemania. Si es compatible con un sistema por encima del PC de IBM, tendrá su alcance más de diez mil aplicaciones y programas, todos los lenguajes de programación, incluyendo el C, Pascal, Modula 2, C++, Basic, Cobol, etc., y una gran variedad de utilidades para el desarrollo de programas profesionales.

... a ver cuando lo vemos por sus fauces.

## OPORTUNIDAD

Por cambio de negocio  
se venden muebles de oficina  
en lote completo o por separado.

(Todo el mobiliario de diez despachos, sala de juntas, etc.)

Teléfono: 231 25 17 y 231 24 02. Preguntar por Angel Rodríguez.



# Este PC de Commodore ha hecho dudar a más de uno.

La oferta del mercado de los ordenadores PC dejaba hasta hoy muy sencilla la elección. Sin embargo, Commodore, líder reconocido en varios sectores de la informática, ha ofrecido una respuesta alternativa que atiende plenamente las exigencias empresariales y de profesionales liberales: su nuevo ordenador PC.

El nuevo Commodore PC dispone de una versatilidad acorde con una tecnología depurada en constante evolución y compatible con el software standard que más le suena.

Estos avances, y un precio realmente interesante, han planteado serios dudas entre los profesionales más cualificados a la hora de elegir un buen PC.

Si usted Commodore, con el mayor número de ordenadores vendidos en el mundo se afianza en el campo empresarial con mucha fuerza.

Si está interesado en conocer más de cerca el nuevo PC de Commodore, pregunte en cualquier concesionario Commodore, le sacará de dudas.

**commodore PC**

**PRINCIPALES CARACTERÍSTICAS**  
- 256 K de RAM de 9 bits - Zócalos para 512 K más - 2 unidades de disquete de 360 K - Disco 10 Mb opcional - Interfases serie y paralelo, incluidas - 5 slots compatibles - Alta resolución incluida - El mejor precio en esta categoría.

  
**commodore**



# DIRECTORIO



- Ordenadores personales Hard y Soft.
- Cursos de Basic.

## RENOVACION EN MARCHA, S.A.

### OFICINAS

C. Española 74-27 tel.  
28003 MADRID  
Teléfono (91) 445 24 78

### REM SHOP-1

C. Gable, 4. 28015 MADRID  
Teléfono (91) 445 28 08

### REM SHOP-2

C. Dr. Casals, 74. 28005 MADRID  
Teléfono (91) 274 56 43

### REM SHOP-3

C. Modesto Lafuente 30  
28007 MADRID  
Teléfono (91) 233 83 18

### REM SHOP- BARCELONA

C. Pelajo 12. Entresuelo  
Teléfono (93) 301 47 05

### REM SHOP- LAS PALMAS

Cra. May de Canemal 46  
Teléfono (928) 23 02 90  
Sólo para el con 25 2 95

## casa de software s.a.

C/ Dragón 272-88-6  
tel. 215-69-52 - 08007 Barcelona

- Software profesional para C-64
- Con distribución productos DIGITAL RESEARCH

## SOFTWARE ESPAÑA

Anda, de Aragón, 19  
14804 La Coruña  
tel. 25 51 72

Especializados  
en software para  
Commodore-64  
Spectrum  
y MSX

## OPORTUNIDAD

Por cambio  
de negocio  
se venden muebles  
de oficina  
en lote completo  
o por separado.

*Todo el mobiliario  
de diez despachos,  
sala de juntas,  
escritura.*

Teléfono: 211 25 17 - 211 24 62  
Presupuesto por Ángel Rodríguez



## HACEMOS FACIL LA INFORMÁTICA

- SINCLAIR • SPECTRAVIDEO
- COMMODORE • DRAGON
- AMSTRAD • APPLE
- SPERRY UNIVAC

En el extranjero: 211 25 17 - 211 24 62  
En el extranjero: 211 25 17 - 211 24 62

En el extranjero: 211 25 17 - 211 24 62  
En el extranjero: 211 25 17 - 211 24 62

En el extranjero: 211 25 17 - 211 24 62  
En el extranjero: 211 25 17 - 211 24 62

En el extranjero: 211 25 17 - 211 24 62  
En el extranjero: 211 25 17 - 211 24 62

En el extranjero: 211 25 17 - 211 24 62  
En el extranjero: 211 25 17 - 211 24 62

## ELECTROAFICION COMPUTER

C. Internet, 104

08011 Barcelona - Tel. 251 76 00 - 89  
4 Gran Via Corts Catalanes, 159  
08017 Barcelona - Tel. 274 21 19

- SILENCIOSA, SILENCIOSA
- SILENCIOSA, SILENCIOSA
- SILENCIOSA, SILENCIOSA
- SILENCIOSA, SILENCIOSA

## RADIO WATT

- ORDENADORES PERSONALES
- ACCESORIOS INFORMÁTICA
- COMPONENTES ELECTRONICOS
- TELECOMUNICACIONES

Paseo de Gracia 126-130  
Tel. 237 11 82 - 08008 BARCELONA

## CENTRO DE INFORMATICA

Las Rozas - Majadahonda  
EMPEZAMOS  
Cursos en BASIC  
cada 15 días

Señores  
todo lo referente al  
COMMODORE 64  
Teléfono: 637 31 51

## ES ELECTRONICA SANDOVAL

DISTRIBUIDORES DE

- ORIC-1
- CASIO FX-200
- ROCKWELL AIM-65
- VIDEO GENIE-EG-2000
- CASIO FX-990SP
- SINCLAIR SPECTRUM
- OSBORNE 1
- DRAGON 32
- NEW BRAIN
- EPSON RX-20

C. SANDOVAL, 3. 4. 6 MADRID 28010  
TELEFONOS 445 18 33 18 33  
TELEX 47706 SAV, E



## VENTAMATIC

• VENTAMATIC: Venta de software y hardware para Commodore 64 y Spectrum. Se ofrecen programas de gestión, contabilidad y otros. Precio muy bajo.

• VENTAMATIC: Venta de software y hardware para Commodore 64 y Spectrum. Se ofrecen programas de gestión, contabilidad y otros. Precio muy bajo.





# Magia

La M4G14 son tracos, la M4G14 es  
directa

La MAGIA es hacer lo que nadie se ha atrevido y resulta ser la fuente más completa de información para la informática práctica.

La **MAGIA** es una sección mensual llena de consejos, trucos, de esto y aquello del mundo del software, hardware y aplicaciones, trucos descubiertos por los demás que hacen que la informática sea más fácil, más divertida e más animada

MAGLA habla de ideas sencillas, programas de una sola línea, subrutinas útiles, hechos de informática poco conocidos y otras cosas de interés.

Los trucos de magia enséñámelos comprobados, pues hay varios incorrectos. ¿Ah? y no nos enseñéis trucos reventados. . . ¡Luzas!



## EL REY Y EL PASTOR

El comensalismo beneficia de un pastel que puede ser grande de amor por a peñes casillas de un laberinto de apóstrifo, o dicho por la segunda y así sea, así sea me!

‘I remain, my dear Mr. Hamilton, your obedient servant’

Particular da sã documentat și însoțite de date care  
sunt în posesia autorității.

1.5, 2.3488, 1.1616, 8.72%, 8.1%, 1981% 1.5

[illegible]

He llamado a todos el día a lo que se esperaba seguir que se había de cumplir.

<sup>1</sup> See, e.g., *United States v. Gurnea*, 199 F.3d 1005, 1010 (9th Cir. 2000).*John Wiley & Sons, Inc.*

© Elsevier B.V. 2005

doi:10.1017/S002229240000199

© 2004 Blackwell Publishing Ltd

## HIGH-QUALITY ENDS

© 2004 American Psychological Association

En mundo este programa de una hora por semana, conversarán p. M. Carli en su sala de reunión. Es un programa generado de computadora, con la ayuda de un sistema de diálogo, tal vez encuentres un amigo en camino.

5. 111111

© 1998 by The McGraw-Hill Companies, Inc. All rights reserved. Printed in the United States of America.

[illegible]

Management: Safety as a Goal

*Pompeii Fabre* 19

*Product\*superscript*

$$t \in \mathbb{R}^n \text{ with } t \cdot t = 1, \quad t \cdot \mathbf{e}_1 = 0,$$

## DESACTIVA TECLAS

Chemos el siguiente texto que he descubierto. Fernando PRINI (1988) se refiere a los textos (sh) y comenta: Es un texto más o menos «a priori» con caracteres programables y en donde el querriendo se dice, o sea, todos.

TABLE 1. HRH in angle, by american ethnic descent

*Joan Reyes-Lamora*

*d. Perenn. Caballero* 25

at 100°C. Samples were



REGISTER LATELY  
CONTINENTAL, S. A.  
AVDA. DE ROMA, 157-59  
08011 - BARCELONA  
TEL. (93) 254 49 58  
Información: Sr. FERRER

**NUESTRA EMPRESA  
AL SERVICIO DE TODOS**

## DEPARTAMENTO COMERCIAL:

- Microprocesadores y ordenadores de gestión (gama Commodore) para el particular y Empresa.
- Programas educativos, de juegos y de diferentes aplicaciones. (Gestión, contabilidad.)

## DEPARTAMENTO DE ESTUDIOS.

- Cursos Programación Basic y Cobol.
- Cursos de grabación.
- Prácticas con ordenadores y micro en la propia empresa.









—Baptist ... a ... of ... ..

Modéliser « qu'on le persuade » « copie » ou « copie »

Qualität des Interviews 0,77 (40 = 74 + 100)

<sup>a</sup>Based on 1000 replications.  $N = 125$  per design.  
<sup>b</sup>For  $\alpha = 0.05$ ,  $\beta = 0.80$ ,  $\sigma^2 = 1$ ,  $\mu = 0$ ,  $\mu_0 = 0.5$ ,  $\mu_1 = 1$ ,  $\mu_2 = 1.5$ .

18. J. J. de P. J. 541-543

El presente artículo se ha escrito con el apoyo de la Subsecretaría de Evaluación Social.

a) Hacia el futuro, ¿cuál es el mayor desafío que enfrentará el mundo de la salud pública? ¿cuál es el mayor desafío que enfrentará el mundo de la salud pública?

La empresa presentada a este programa fue Abanda y Cia S. de RL de CV, una prima. DORWIN se me ocurrió llamar así a la empresa en un momento de la entrevista, con la idea de que se formaría un triángulo, como cuando se ingresa por un cierto número de datos, para hacerlos.



La relación entre cada individuo y su entorno es diferente, por lo tanto, la deriva a la izquierda de un individuo es diferente a la deriva a la izquierda de otro individuo, en la misma especie, que a su vez, es diferente a la deriva a la izquierda de un individuo de otra especie.

Consideremos, para el primer caso, el caso  $n = 2^k$ . En primer lugar, se puede escribir  $g = 2^k$  como suma de potencias de 2 de la siguiente manera:

En la primera relación, con 4 buses cada litro. Al bajar al segundo nivel, se aumentan dos del nivel 2 a cada gasolinero, pero en el punto 3, como se muestran 2 buses por gasolinero de cada uno de los puntos que había en el nivel 2, por tanto, resulta 4 buses.

Los papeles comunes incluyen papeles que provienen de dos papeles que tienen en su



La derivación que se otorga a PROBABILIDAD en los cuatro nodos finales será 1, 1, 1 y 1.

Il ha cinque fili di protezione con 9 paia di fili totali in: Monofila in nylon, di colore: 1, 3, 28, 56, 70, 56, 28, 3, 1.

Estas destribas, y sus harnadas, bu ornales se pueden coonociar en:  
 Harnada de los coll. Triunfo, de harnaga, de Pissal, en el cual  
 cada una de las harnadas tiene un ornales, y cada uno de los ornales se  
 ubica en su harnada, los dos que hay sobre el



La teoría del postgradismo resultante es parodia a la llamada campaña de Luján, así a forma de encuesta:

Expuso que el programa se gusta a pesar del "bulto" mencionado por los puntos en la carta.

[illegible]

 (952) 27 30 43 - OFERTA ABRIL

- **SINCLAIR: "QL" SOLO 86.500 PESETAS**  
incluido: 4 programas, manual en español, garantía 6 meses.
- **COMMODORE: UNIDAD DE DISCO 148 KB PARA C-64**  
con todos sus programas disponibles.
- **AMSTRAD: AHORA ENTREGAMOS GRATIS 12 PROGRAMAS**  
**CON CADA AMSTRAD.**
- **COMERCIO: ::LIMITADA!!**

incluido: 4 programas, manual en español, garantía 6 meses.

● COMMODORE: UNIDAD DE DISCO 148 KB PARA C-64

con todos sus programas disponibles.

● **AMSTRAD: AHORA ENTREGAMOS GRATIS 12 PROGRAMAS CON CADA AMSTRAD.**

● COMERCIO: ::LIMITADA!!

LOVERCIC

C/Compositor Llorenç Ruiz, 1  
29007 MALAGA - Tèlex 23400 Ccc

29007 MALAGA - Telex 77400 Coca









```

.. C000 00 00 000 000
.. C001 20 00 000 000
.. C002 00 00 000 000
.. C003 00 00 000 000
.. C004 00 00 000 000
.. C005 00 00 000 000
.. C006 00 00 000 000
.. C007 00 00 000 000
.. C008 00 00 000 000
.. C009 00 00 000 000
.. C010 00 00 000 000
.. C011 00 00 000 000
.. C012 00 00 000 000
.. C013 00 00 000 000
.. C014 00 00 000 000
.. C015 00 00 000 000
.. C016 00 00 000 000
.. C017 00 00 000 000
.. C018 00 00 000 000
.. C019 00 00 000 000
.. C020 00 00 000 000
.. C021 00 00 000 000
.. C022 00 00 000 000
.. C023 00 00 000 000
.. C024 00 00 000 000
.. C025 00 00 000 000
.. C026 00 00 000 000
.. C027 00 00 000 000
.. C028 00 00 000 000
.. C029 00 00 000 000
.. C030 00 00 000 000
.. C031 00 00 000 000
.. C032 00 00 000 000
.. C033 00 00 000 000
.. C034 00 00 000 000
.. C035 00 00 000 000
.. C036 00 00 000 000
.. C037 00 00 000 000
.. C038 00 00 000 000
.. C039 00 00 000 000
.. C040 00 00 000 000
.. C041 00 00 000 000
.. C042 00 00 000 000
.. C043 00 00 000 000
.. C044 00 00 000 000
.. C045 00 00 000 000
.. C046 00 00 000 000
.. C047 00 00 000 000
.. C048 00 00 000 000
.. C049 00 00 000 000
.. C050 00 00 000 000

```

```

.. C051 00 00 000 000
.. C052 00 00 000 000
.. C053 00 00 000 000
.. C054 00 00 000 000
.. C055 00 00 000 000
.. C056 00 00 000 000
.. C057 00 00 000 000
.. C058 00 00 000 000
.. C059 00 00 000 000
.. C060 00 00 000 000
.. C061 00 00 000 000
.. C062 00 00 000 000
.. C063 00 00 000 000
.. C064 00 00 000 000
.. C065 00 00 000 000
.. C066 00 00 000 000
.. C067 00 00 000 000
.. C068 00 00 000 000
.. C069 00 00 000 000
.. C070 00 00 000 000
.. C071 00 00 000 000
.. C072 00 00 000 000
.. C073 00 00 000 000
.. C074 00 00 000 000
.. C075 00 00 000 000
.. C076 00 00 000 000
.. C077 00 00 000 000
.. C078 00 00 000 000
.. C079 00 00 000 000
.. C080 00 00 000 000
.. C081 00 00 000 000
.. C082 00 00 000 000
.. C083 00 00 000 000
.. C084 00 00 000 000
.. C085 00 00 000 000
.. C086 00 00 000 000
.. C087 00 00 000 000
.. C088 00 00 000 000
.. C089 00 00 000 000
.. C090 00 00 000 000
.. C091 00 00 000 000
.. C092 00 00 000 000
.. C093 00 00 000 000
.. C094 00 00 000 000
.. C095 00 00 000 000
.. C096 00 00 000 000
.. C097 00 00 000 000
.. C098 00 00 000 000
.. C099 00 00 000 000
.. C100 00 00 000 000

```

# GLOSARIO

**Data conversion** - Conversión de datos. El proceso de transformación de los datos de un sistema de representación a otro.

**Data density** - Densidad de datos. El número de bytes por pulgada (tipo de almacenamiento) en un cinta magnética.

**Data error** - Error en disco. La distorsión de datos como se esperaba, normalmente se produce en los cálculos aritméticos y puede ser causado por los errores en los cálculos o de memoria.

**Data files** - Archivo de datos. Un grupo de datos relacionados con un objeto real o ficticio determinado, por ejemplo, es de una compañía.

**Data flowchart** - Diagrama de flujo de datos. Es un organigrama o diagrama de flujo que representa las operaciones a que son sometidos los datos en un programa para resolver un problema determinado.

**Data integrity** - Integridad o seguridad de datos. La seguridad de los datos frente a su destrucción accidental o intencional, o de su uso para evitar el acceso a personas no autorizadas.

**Data library** - Biblioteca de datos. Es un conjunto de archivos de datos relacionados entre sí. Por ejemplo los datos de un control de stock.

**Data management** - Gestión de datos. La función de controlar la entrada, análisis, almacenamiento, almacenamiento y recuperación de datos.

**Data processing (DIP)** - Proceso de datos. La realización sistemática de operaciones en el tratamiento de los datos.

**Debug** - Depurar. La detección, corrección y eliminación de errores en un programa de ordenador.

**Decimal alignment** - Alineamiento decimal. El formato en el que los datos decimales en los números representados en pantalla o en impresora.

**Default** - Por falta. Es un valor automático que toma una variable en caso de falta o omisión.

**Degradation factor** - Factor de degeneración. La medida del deterioro de los datos en un medio magnético en función del tiempo y el uso.

**Delay** - Retardo. El tiempo que se retarda o detiene un proceso.

**Delete (DEL)** - Borrar. Quitar un carácter, dato o información de la memoria, programa o unidades auxiliares de memoria.

**Delimiter** - Delimitador. El carácter que indica la separación de los datos en una cadena string. Puede ser un signo de puntuación, un código que no aparece normalmente.

**Designate** - Designar para indicar o catalogar el tipo de información que contiene un archivo, generalmente se coloca en el primer registro de cabecera.

**Desktop computer** - Ordenador de sobremesa. Es otra palabra utilizada para referirse a un microcomputador.

**Development system** - Sistema de desarrollo. Un sistema de microcomputador que sirve para facilitar el desarrollo y evaluación de programas y entornos software y hardware.

**Development time** - Tiempo de desarrollo. El tiempo que se tarda en preparar y depurar las rutinas o el hardware, es sinónimo de tiempo de programación.

**Development tools** - Herramientas de desarrollo. Son aquellos programas de unidades que facilitan el diseño de programas o circuitos.









## EXCURSION POR LA MICROPROGRAMACION

# Rincón del Código Máquina

**E**l mundo como tal no ha terminado, pero sugiero comenzar a escribir rutinas útiles de código máquina en este "Rincón del Código Máquina", en el veterano y antiguo sistema que, con muchas de mis rutinas programadas, nos permitieron experimentar la velocidad de sus funciones que son más lentas a veces, pero nos permiten realizar sin recurrir al código máquina.

En este número publicamos una rutina que permite copiar el RAMBL de ROM a RAM, de este modo podremos almacenar las direcciones de memoria que a nosotros nos interesa. La rutina ocupa solamente las posiciones de memoria \$C000 a \$C01B, por lo tanto 22 bytes de programa, podemos hacer en menos de un segundo lo que tardaría más de medio minuto en Basic.

Para copiar el intérprete de ROM a RAM basta con incorporar a nuestro programa una línea en que hagamos SYS 44152.

Después de copiar la rutina, ya tenemos el Basic en la RAM, ya que se encuentra bajo la ROM, y si queremos podemos hacer POKE 1,54 para cambiar la configuración de memoria con la que trabaja la CPU. Esta rutina puede utilizarse en nuestros propios programas sin tener que realizar las siguientes:

El programa en código máquina que ocupa las direcciones de memoria \$C011 a \$C01B puede utilizarse también por sí solo, ya que no utiliza las restantes rutinas. Este programa permite añadir un nuevo comando a nuestro Basic, el PRINT AL. Aunque no lo anade realmente a la lista de comandos del intérprete, lo podemos simular con el comando



SYS 44152, columna, fila, "texto". También se puede aceptar el tamaño del programa y la dirección, el valor 44152 a una variable y ganamos cuatro bytes cada vez que debemos llamar a la rutina PRINT AL, esto se llama así:

```
10 P = 44152
```

```
1000 SYS P, columna, fila, "funciona"
```

```
1500 SYS P, columna, fila, al
```

Sugiero que este nuevo comando sirva de ayuda a muchos que están acostumbrados a utilizarlo en otros ordenadores y facilitara la tarea de conseguir una buena representación en pantalla para los programas.

A muchos les gustará disponer de instrucciones GOTO y GOSUB calculadas, pues están acostumbrados a utilizarlas con otras versiones de Basic, les parece difícil de utilizar una rutina de comandos GOSUB y GOTO. Para esto se diseñó la rutina que ocupa las direcciones \$C011 a \$C01B. Esta rutina necesita que esté en su lugar la de copia de

ROM a RAM (la primera rutina), ya que debe copiar el intérprete Basic y cambiar una instrucción en la rutina que se encarga de los comandos GOTO y GOSUB para que al ejecutarlos pase por la rutina (NUEVO GOTO).

Para activar los GOTO y GOSUB calculados es necesario teclear el comando

```
SYS 44137
```

y poder comprobar que todo va bien con un programa como el siguiente:

```
10 L=40: GOTO L
20 PRINT "ESTO NO FUNCIONA"
30 END
40 PRINT "ESTO FUNCIONA BIEN"
```

En lugar de hacer GOTO 1 se pueden teclear fórmulas completas dentro del GOTO, por ejemplo GOTO 2+(5\*11/2)/3.

Otro comando sobre el que hemos recibido varias peticiones por carta es el RESTORE N, de modo que si debemos repetir varias veces una sucesión de datos en un programa podemos utilizarlo sin necesidad de leer todas las líneas DATA hasta encontrar la línea que nos interesa.

Para añadir este comando al intérprete Basic, necesitamos alterar la rutina que ocupa las posiciones de memoria \$C009 a \$C011. Para que este programa funcione es necesario que estén en sus respectivas posiciones de memoria las rutinas de los GOTO y la de copia de memoria, ya que son llamadas tanto al activarlo como al ejecutarlo.

Para añadir este comando se nos necesitan teclear SYS 44152.

No es necesario haber activado la rutina anterior, ya que sólo

## EXCURSION POR LA MICROPROGRAMACION

ultimo SYS la activara automáticamente, añadiendo los comandos GOTO N, GOSUB N y RESTORE N.

Para terminar por este mes, la rutina que ocupa las posiciones \$C072 a \$C181, está destinada a "mirar" el contenido de la RAM que se encuentra escondida debajo de la ROM del KERNAL y del BASIC. Insistiblemente fue pensada para ser utilizada solamente como PRINT PEEK (DIRECCION), pero luego le quite la llamada a la rutina PRINT, y añadí las instrucciones que ocupan las direcciones \$C182 a \$CDA2.

Esta última rutina permite asignar el valor del "PEEK" a una variable de tipo numerico. Para hacer un PEEK bajo la ROM debemos incluir la siguiente línea en Basic:

```
SYS 44243, variable, direccion
Por tanto si sabemos el valor
del comando de la direccion
65000 de RAM, podríamos hacer:
10 SYS 44243,A,65000
20 PRINT A
```

La dirección puede ser también una variable, pues la rutina del PEEK bajo ROM se encarga de "evaluar" la dirección por medio de la llamada a la rutina \$V AARG del intérprete Basic (SYS ISR V AARG).

\*\*\*[na observación importante: la variable que se utiliza para ver el contenido de la RAM oculta debe ser del tipo de coma flotante, ya que la rutina de código máquina no comprueba si lo es o no, y en caso de utilizar una variable entera (por ejemplo A) ]

**S**upongo que este nuevo comando servirá de ayuda a muchos que están acostumbrados a utilizarlo en otros ordenadores y facilitará la tarea de conseguir una buena representación en pantalla para los programas.

dada valores que nada tienen que ver con la realidad.

Espero que sean útiles para todos las rutinas que he recopilado, y sobre todo la última para aquellos que, pese a tener 10K RAM libres, les sigue pareciendo poco y quieren disponer de la RAM que les han robado el intérprete y el kernel. Otra aplicación de esta rutina sería para proteger vuestros programas, ya que podáis examinar datos bajo la ROM y hacer que no funcionaran los programas si los datos no fueran correctos.

Además de las posibles aplicaciones que les podáis dar a estas rutinas, creo que la principal es la de aprender observando el modo de llamar las rutinas del intérprete, y poderlas intentar utilizar las en vuestros programas. Por ejemplo la rutina \$B SCAM (\$A) ID sirve para comprobar si el siguiente carácter del programa basic es una coma, y en caso contrario da un "SYNTAX ERROR".

La rutina \$V AARG (\$ADRA)

evalúa el argumento de una función o comando Basic, y el resultado lo guarda en el acumulador de coma flotante numero uno.

La rutina \$AC ADDR (\$B7) sirve para convertir un numero entero situado en el acumulador numero uno a formato binario de 16 bits, ya que aunque trabajemos con numeros enteros en el programa basic, el ordenador los trata como numeros de coma flotante para todos sus calculos aritmeticos.

La rutina \$B00B sirve para encontrar la dirección de memoria que ocupa una variable Basic, y devolver la dirección física de la variable en los registros A y Y de la CPU.

La rutina \$B191 pasa al acumulador numero uno el numero binario contenido en los registros A y Y.

La rutina \$B100 pasa los datos del acumulador numero uno a la variable señalada por el contenido de las posiciones \$44 y \$4A.

Para aquellos que quieran utilizar estas rutinas en la memoria del ordenador y no dispongan de un ensamblador, hemos preparado un programa en basic que contiene las rutinas de código máquina en forma de segmentos \$DA1A con un basic que las va leyendo y colocando en sus posiciones respectivas, es el estado 2.

Para facilitar la detección de errores al teclearlo he añadido también la suma de control del programa (en numero 10, página 161).

### PROGRAMAS UTILITARIOS

ESTE ES UN CONJUNTO DE RUTINAS  
CON DIVERSAS FUNCIONES UTILES  
PARA INCORPORARLAS A VUESTROS  
PROPIOS PROGRAMAS.  
(C) DIEGO ROMERO 1985  
COMMODORE WORLD-C.H. SPAIN

```
1000: C000
1050: C000
1100: C000
1110: C000
1120: C000
1130: C000
1140: C000
1150: C000
1160: C000
```

```
BASIC = B0000
VECTOR = BFC
VRESTO = B0B24
AUTOIT = B0B00
PARIZO = B0B77
PARGER = B0BFA
BUSCOM = B0BFD
PLOT = B0BFB
PRINT = B0B00
VECTOR LIBRE
VECTOR DE RESTORE
ROUTINA GOTO
BUSCA 1
BUSCA 2
BUSCA 3
BUSCA COMA
LEA/FIJA POSICION
ROUTINA PRINT
```

# EXCURSION POR LA MICROPROGRAMACION

```

1170: C000
1180: C000
1190: C000
1195: C000
1197: C000
1200: C000
LEEDYT = #0790 ; LEE ARGUMENTO (B-255)
EVNARG = #0000 ; EVALUA ARGUMENTO
FACROR = #0777 ; PASA FAC A ADDRESS
CALROR = #0613 ; CALCULA DIRECCION DE LINEA
ILEORL = #0246 ; ILEGE QUANTITY
= #C000

```

## ROUTINA DE COPIA DEL BASIC

```

1210: C000 A2 R0
1220: C002 R0 R0
1230: C004 R4 FF
1240: C006 R4 FE
1250: C008 A2 20
1260: C00A R1 FE
1270: C00C R1 FE
1280: C00E C0
1290: C010 C0 F0
1300: C012 R6 FF
1310: C014 C0
1320: C016 C0 F4
1330: C018 C0
R0R0S LDX #BASIC ; DIRECCION INICIAL
LDY #BASIC ; DE COPIA ROM-RAM
STX VECTOR+1
STY VECTOR
LDX #420 ; PAGINAS A COPIAR
LDR (VECTOR),Y
STR (VECTOR),Y ; COPIA UN BYTE
INY
BNE COPIA ; SIGUIENTE BYTE
INC VECTOR+1
DEX ; CUENTA PAGINAS
BNE COPIA ; SIGUIENTE PAGINA
RTS

```

## ROUTINA PRINT-RT

```

1340: C017 20 F0 R6
1350: C01A 20 R6 B7
1360: C01C R6 FE
1370: C01F 20 F0 R6
1380: C022 20 R6 B7
1390: C025 R4 FE
1400: C027 10
1410: C029 20 F0 FF
1420: C02B 20 F0 R6
1430: C02D 20 F0 R6
1440: C02F 4C R0 R0
JSR BUSCOM ; NUMERO COLUMNA
JSR LEEDYT+3 ; GUARDA EL VALOR
STX VECTOR
JSR BUSCOM ; NUMERO DE LINEA
JSR LEEDYT+3 ; RECUPERA COLUMNA
LDY VECTOR
CLC ; COLOR EL CURSOR
JSR PLOT ; BUSCA CORA
JSR BUSCOM ; VA A PRINT NORMAL
JMP PRINT

```

## ROUTINA DOTS-GOSUB CALCULADOS

```

1450: C031 20 R0 R0 HUSGOT
1460: C034 4C F7 B7
1470: C037 20 R0 C0 CAMBIA
1480: C03A R0 C0
1490: C03C R2 11
1500: C03E R0 A2 R0
1510: C041 R0 A1 R0
1520: C044 R0 36
1530: C046 R0 R0
1540: C048 60
JSR EVNARG ; LEE ARGUMENTO
JMP FACROR ; LO CONVIERTE EN DIRECCION
JSR R0R0S ; PASA BASIC A RAM
LDX #HUSGOT
LDR RUTGOT+2 ; PONE NUEVO VECTOR
STX RUTGOT+1 ; EN RUTINA DOTS
LDX #54
STR #01 ; CAMBIA A RAM
RTS

```

## ROUTINA PARG RESTORE HH

```

1510: C049 R0
1515: C04A F0 15
1520: C04C 20 31 C0
1530: C04F 20 13 R0
1540: C052 30
1550: C053 R0 5F
1560: C055 E9 R0
1570: C057 R4 60
1580: C059 R0 R0
1590: C05B R0
1600: C05C R0 41
1610: C05E R4 42
1620: C060 R0
1630: C061 4C 10 R0 HRESTO
1640: C064 20 37 C0 ACTIVA
1650: C067 R0 C0
1660: C069 R2 49
1670: C06B R0 25 R0
1680: C06E 24 R0
HRESTO NOP
BEG HRESTO ; NUEVO RESTORE
JSR HUSGOT ; CALCULA LINEA DESTINO
JSR CALROR ; LA PASA A DIRECCION
SEC
LDR #5F ; LEE DE DIRECCION
SBC #1 ; LE RESTA 1
LDY #60 ; HSB DE DIRECCION
BCC ETIQU1 ; NO HAY HRESTO
DEY
STR #41 ; PUNTERO DE DATOS
STY #42
RTS
JMP #010 ; ANTIQUO RESTORE
JSR CAMBIA
LDX #HRESTO ; HSB DEL NUEVO RESTORE
LDX #HRESTO ; LEE
STX HRESTO+1
STX HRESTO

```

# EXCURSION POR LA MICROPROGRAMACION

1790: C071 60

RTS

RUTINA PEEK BAJO ROM

```

1800: C072 20 FD RE PEEKROM JSR BUSCOM  BUSCAR COMP
1800: C075 20 BR RD JSR ENVARO  LEER LA DIRECCION
1800: C076 20 F7 B7 JSR PROCOR  LEER PASA A 16 BITS
1910: C070 78 SETI  INTERRUPCIONES
1920: C07C F5 01 LDR  #1  CONFIGURACION ACTUAL
1930: C07E 40 AND  #0  LEER DUNDO EN EL STACK
1940: C07F 29 FD STR  #1  QUITA ROM
1950: C083 05 01 LDR  #0  LEER EL BYTE DE RAM
1960: C083 02 00 LDR  (R14,X)  LO PASA AL REGISTRO Y
1962: C085 R1 14 TRY  RECUPERA CONFIGURACION
1964: C087 60 PLA  REACTIVA INTERRUPCIONES
1966: C088 60 STR  #1
1968: 1800 05 01 CLT
1970: C08E 50 RTS
2000: C09C 60
    
```

RUTINA VARIABLE=PEEK(ROM)

```

2010: C09D 20 FD RE JSR BUSCOM  BUSCAR COMP
2020: C09E 20 00 00 JSR #0000  BUSCAR LA VARIABLE
2030: C093 05 49 STR  #49  GUARDAR LA DIRECCION
2040: C095 04 40 STY  #40  DE LA VARIABLE
2050: C097 20 72 C0 JSR PEEKROM  PEEK BAJO ROM
2060: C09A 00 00 LDR  #0
2100: C09C 20 91 01 JSR #0091  16 BITS AL ACUM 1
2110: C09F 20 00 00 JSR #0000  PAC A VARIABLE
2120: C0A2 60 RTS
    
```

## PROGRAMAS UTILITARIOS

```

10 I=49152
20 READ R:IF R=256 THEN 40
30 POKE I,R:I=I+1:GOTO 20
40 IF S<21286 THEN PRINT "ERROR EN DATOS"
50 END
49152 DATA 162,160,160,8,134,255,132,254
49160 DATA 162,32,177,254,145,254,200,200
49160 DATA 249,238,235,202,200,244,36,32
49170 DATA 253,174,32,158,103,134,254,24
49180 DATA 253,174,32,158,103,164,254,24
49190 DATA 32,240,255,32,253,174,76,160
49200 DATA 179,32,138,173,76,247,183,32
49200 DATA 8,192,169,192,162,49,141,162
49210 DATA 160,142,161,160,169,54,133,1
49224 DATA 96,234,240,21,32,49,192,32
49232 DATA 19,166,56,165,95,233,1,164
49240 DATA 96,176,1,136,133,65,132,66
49248 DATA 96,76,29,160,32,55,192,169
49256 DATA 192,162,73,141,37,160,130,173
49264 DATA 160,56,32,253,174,32,130,173
49272 DATA 32,247,103,120,165,1,72,41
49280 DATA 253,133,1,162,8,161,20,160
49288 DATA 104,130,1,80,96,32,253,174
49296 DATA 32,139,176,133,73,132,74,32
49304 DATA 114,192,169,8,32,145,179,32
49312 DATA 200,107,56,256
    
```

## PROGRAMAS UTILITARIOS

### SUMA DE CONTROL DEL CARGADOR BASICO

10	0	20	160	30	136
40	105	50	120	49152	51
49160	112	49160	67	49176	68
49104	72	49192	13	49200	22
49200	20	49216	21	49224	171
49232	103	49240	179	49240	200
49256	13	49264	20	49272	163
49200	155	49208	174	49296	222
49304	222	49312	189		





```

2
*****
10
10  PAQUETE PARA GRAFICOS EN ALTA
10
10  RESOLUCION EN EL COMODORE 64
10
10  MICROELECTRONICA Y CONTROL S.R.L.
10
*****
1
1
411  6000  ORIGIN  =  6000
1
1
***** RUTINAS DEL SISTEMA *****
1
1
471  6000  ERROK  =  4047  ;EJECUCION MENSAJE DE ERROR.
481  6000  EVALOP  =  4048  ;EVALUAR EXPRESION.
491  6000  CHACON  =  4049  ;BUSCAR LA CORR.
501  6000  PLTPIX  =  404A  ;CONVERTIR A FIJO EN Y (GRAF.) Y A
;ALT.
1
1
***** VECTORES *****
1
1
501  6000  ERVEK  =  4000  ;RUTINA DE ERROR.
511  6000  BARRY  =  4002  ;INICIO DE BASIC (MARIO).
1
1
***** VARIABLES DE VIDEO *****
1
1
531  6000  VIC     =  4000  ;DIRECCION DEL CHIP DE VIDEO.
541  6000  VICRCL =  VIC+17 ;CONTROL DE MODO.
551  6000  VICREG  =  VIC+24 ;CONTROL DE MEMORIA.
1
1
571  6000  SCREEN  =  4040  ;REN. DE PANTALLA.
581  6000  SCRENO  =  SCREEN+99 ;MULTIP. DIRECCION DE LA PANTALLA.
591  6000  BASE    =  4000  ;INICIO DE ALTA RES.
601  6000  HALST   =  BASE+7999 ;ULTIMA POSICION.
611  6000  RAM     =  403C  ;BUFFER DEL CASSETTE.
1
1
***** PAGINA CERO *****
1
1
771  6000  BYT     =  4F0   ;PUNTERO DE BYTE.
1
1
881  833C  **      =  83m
1
1
831  833E  X1      =  4+2   ;COORDENADA X (0 - 319).
841  8340  X2      =  4+2
851  8342  Y1      =  4+2   ;COORDENADA Y (0 - 199).
861  8344  Y2      =  4+2
871  8346  BIT0    =  4+1   ;BIT DEL BYTE.
881  8347  DELT0   =  4+2   ;X2-X1.
891  8349  DELT1   =  4+2   ;Y2-Y1.
901  834B  E        =  4+2
911  834D  T        =  4+2
921  834F  C        =  4+2
931  8350  I        =  4+1   ;PUNTERO DE DIRECCION.
941  8352  TEMP    =  4+2
951  8354  ERVEK   =  4+2   ;CONTIENE EL VECTOR DE ERROR.
1
1
***** CONSTANTES *****
1
1
101  8354  XPRK    =  320
102  8354  YPRK    =  200
103  8354  CULS    =  40    ;REN. DE COLUMNAS POR FILA.
104  8354  COLOR   =  850    ;COLOR DEL FONDO/TRAZO (VERDE ORO)

```



```

LIN DIR. HEXDEC. ETIQUETA COD OPER PCOM. PRO: "ALTA RESOLUCION"
HECRO).

107: 6000          ** 00101H

TABLA DE SALTO.

112: 6000 4C 50 62 JINIT JMP HINDET INICIALIZAR.
113: 6003 4C 80 62 JMBST JMP HMBST POLVER A BASIC.
114: 6006 4C 50 62 JCLR JMP CLSRA BORRAR PANTALLA.
115: 6009 4C 09 60 JDRWA JMP VECPLT DIBUJAR RECTA.
116: 600C 4C 5C 62 JSETPX JMP SETPIX AUMENTAR PUNTO (PIXEL).

ROUTINA HADOR - DORAS (CADA X / 2 BYTES)
(Y COORD Y : 1 BYTE)
(CALCULAR LA DIRECCION DEL BYTE Y EL
BIT DENTRO DE EL BITIND).

ALTERA X, DEJA Y=0.

SALTAR AQUI DESDE BASIC.

129: 600F 20 E2 62 HADROV JSR GETVAL JOTENER X0.
129: 6012 6C 3C 03 STY X1
130: 6015 6C 3E 03 STY X2 PHAN ANCHO.
131: 6018 80 30 03 STA X1+1
132: 601B 80 3F 03 STA X2+1
133: 601E 20 E2 62 JSR GETVAL JOTENER Y0.
134: 6021 6C 40 03 STY Y1
135: 6024 6C 42 03 STY Y2
136: 6027 80 43 03 STA Y2+1
137: 602A 20 E6 61 JSR ANCHO

SALTAR AQUI SI X1 Y1 TIENEN VALOR.

141: 602D 80 80 HADRO LDR #0 BYTE ALTO A CERD.
142: 603F 05 FE STA BYT+1
143: 6041 38 SEC ICALC. 199-Y1
144: 6042 83 C7 LDR #YHAN-1
145: 6044 8D 40 03 SBC Y1
146: 6047 4B PHA GUARDAR RESULTADO EN STACK.
147: 6049 23 FD RND #0F0 ICALC. NUM. FILA.
148: 604B 08 RSL #0 INULT. POR 2.
149: 604D 2C FE ROL BYT+1
150: 604F 0A RSL # INULT. POR 4.
151: 6051 24 FE ROL BYT+1
152: 6053 0A RSL # INULT. POR 8.
153: 6055 26 FE ROL BYT+1
154: 6057 4B PHA GUARDAR EN STACK.
155: 6059 8D 50 03 STA TEMP Y EN TEMP.
156: 605B 85 FE LDR BYT+1
157: 605D 8D 51 03 STA TEMP+1 EN TEMP BYT
158: 605F 4B PLA RESTAURAR A
159: 6061 0A RSL # INULT. POR 16
160: 6063 26 FE ROL BYT+1
161: 6065 0A RSL # INULT. POR 32
162: 6067 24 FE ROL BYT+1 ICARRY RUN 0
163: 6069 8D 50 03 ROL TEMP ICALC. 32+0 = 40
164: 606B 05 FD STA BYT EN BYT
165: 606D 85 FE LDR BYT+1
166: 606F 8D 51 03 ROL TEMP+1
167: 6071 85 FE STA BYT+1
168: 6073 8D 3C 03 LDR X1 JSUM CRR.
169: 6075 23 FD RND #0F0
170: 6077 45 FD ROL BYT
171: 6079 85 FD STA BYT
172: 607B 8D 30 03 LDR X1+1
173: 607D 45 FE ROL BYT+1
174: 607F 85 FE STA BYT+1 ICARRY RUN 0
175: 6081 4B PLA JSUM LINER
176: 6083 23 07 RND #7 ENHANCERADO BITS ALTOS
177: 6085 45 FD ROL BYT
178: 6087 85 FD STA BYT
179: 6089 85 FE LDR BYT+1
180: 608B 45 20 ROL # 0 PAGE
181: 608D 85 FE STA BYT+1

```



```

LIN  DIF.  MENDEC.  ETIQUETA  COD  OPER  ;COM. PRO: "ALTA RESOLUCION"
1001  0001  00  0  03  LDA  X1  ;CALC. BITHO
1003  0002  20  07  AND  #7  ;CON BITS BAJOS
1004  0001  00  TAB  ;INDICE DE LA TABLA
1005  0002  00  20  03  LDA  BITHO,X
1006  0005  00  44  03  STA  BITHO
1007  0000  00  RTS  ;BYT Y BITHO CALCULADOS

```

```

;
;***** PASTPLOT *****
;
;SUBROUTINA GRAFICA PARA DIBUJAR UNA RECTA
;EN LA MEMORIA DE ALTA RESOLUCION
;
;SE ENTRA CON VALORES PARA X1,Y1 Y X2,Y2
;Y EL PRIMER PUNTO ENCENDIDO, ESTO ES
;BYT, BYT+1, Y BITHO ESTAN CALCULADOS
;POR LLAMAR A LA RUTINA PLOT
;
;DEJA Y=0, X ALTERADO Y TESTEA EL REBOSAMIENTO
;
;***** VECPLT *****
;
;SALTAR AQUI DESDE BASIC
;

```

```

2001  0000  20  E2  02  VECPLT  JSR  GETYRL  ;OBTENER COORD. X
2003  000C  0C  3E  03  STY  Y2
2005  0000  00  07  03  STA  X2+1
2006  0002  20  E4  02  JSR  GETYRL  ;OBTENER COORD. Y
2008  0005  3C  4E  03  STY  Y2
2009  0000  00  03  03  STA  Y2+1
;
;SALTAR AQUI SI X2,Y2 ESTABLECIDOS
;
2101  0000  20  C6  03  VECPLT  JSR  BNDCHK  ;VER. PEDIOS. DE X2,Y2
2103  000C  30  SEC  ;FORMAR DELTX (CON SIGNO)
2105  0000  00  3E  03  LDA  Y2
2107  0004  00  3C  03  SEC  X1
2109  0005  00  45  03  STA  DELTX
2111  0000  00  30  03  LDA  X2+1
2113  0000  00  30  03  SEC  X1+1
2115  0000  00  46  03  STA  DELTX+1
2117  0001  30  SEC  ;FORMAR DELTY (CON SIGNO)
2119  0002  00  42  03  LDA  Y2
2121  0005  00  40  03  SEC  Y1
2123  0000  00  47  03  STA  DELTY
2125  0000  00  47  03  LDA  Y2+1
2127  0000  00  41  03  SEC  Y1+1
2129  0001  00  48  03  STA  DELTY+1
2131  0004  00  3E  03  LDA  X2
2133  0007  00  3C  03  STA  X1
2135  0000  00  3F  03  LDA  X2+1
2137  0000  00  3D  03  STA  X1+1
2139  0000  00  42  03  LDA  Y2
2141  0003  00  40  03  STA  Y1
2143  0006  00  43  03  LDA  Y2+1
2145  0009  00  41  03  STA  Y1+1
;
;
;AHORA TENEMOS DELTX,DELTY (CON SIGNO)
;
;**** MOVE ****
;
;DADOS DELX,DELTY
;DIBUJAR/MOVER PARA LA MEJOR RECTA
;

```

```

2401  000C  00  00  MOVE  LDA  #0  ;DETERMINAR EL OCTANTE
2403  000E  00  4F  03  STA  I
2405  0001  2C  46  03  BIT  DELTX+1  ;VER SI DELTX < 0
2407  0004  10  17  BPL  MV1
2409  000C  00  45  03  LDA  DELTX  ;COMBINAR SIGNO
2411  0003  20  E7  01  JSR  COMPH
2413  000C  00  45  03  STA  DELTX
2415  0007  00  46  03  LDA  DELTX+1
2417  0000  20  E9  01  JSR  COMPH
2419  0005  00  46  03  STA  DELTX+1
2421  0000  00  02  LDA  #2
2423  0000  00  4F  03  STA  I
2425  000D  2C  40  03  BIT  DELTY+1  ;VER SI DELTY < 0
2427  0000  10  10  BPL  MV2

```



LTN DIR. HEXDEC. ETIQUETA CUO OPER JCOP. PROI \*ALTA RESOLUCION \*

```

2601 6162 H0 47 03 LDR DELTY
2611 6165 20 67 01 JSR CORPL
2621 6168 60 47 03 STR DELTY
2631 6169 H0 48 01 LDR DELTY+1
2641 616C 20 68 01 JSR CORPM
2651 6171 50 48 03 STR DELTY+1
2661 6174 19 CLC
2671 6175 H0 48 03 LDA I
2681 6176 09 04 ROR #4
2691 6178 60 48 01 STR I
2701 617D 4E 45 03 HV2 LDR DELTX JVER DELTX-DELTY
2711 617F 6C 47 03 CPX DELTY JCMPLY I PARA BYTE BAJO
2721 6182 60 46 01 LDR DELTX+1 JBYE ALTO
2731 6184 08 THY JYDELTX
2741 6187 6D 46 03 SBC DELTY+1
2751 6188 10 18 BPL HV3
2761 618C 60 47 03 LDR DELTY JINTERCAMBIAH DELTX.V
2771 618F 60 45 03 STR DELTX
2781 6192 60 46 03 LDR DELTY+1
2791 6195 60 46 03 STR DELTX+1
2801 6198 67 40 STX DELTY
2811 619B 6C 48 03 STY DELTY+1
2821 619E 19 CLC
2831 619F H0 47 03 LDR I
2841 61A2 09 06 ROR #6
2851 61A4 60 47 03 STR I
2861 61A7 H0 45 03 HV3 LDR DELTX JCALC, E=-DELTX/2
2871 61A8 20 67 01 JSR CORPL
2881 61AD 6E 45 03 STX E
2891 61B0 H0 46 03 LDR DELTX+1
2901 61B3 10 68 01 JSR CORPM
2911 61B6 6D 46 03 STR E+1
2921 61B9 38 SEL JVER SI NEGATIVO
2931 61BA 30 01 BHI HV4
2941 61BC 19 CLC
2951 61BD 6E 46 03 HV4 ROR E+1 JDIVIDIR POR 2
2961 61C0 6E 45 03 ROR E
2971 61C3 00 08 LDM #0 JYH
2981 61C6 0C 40 03 STY C JCONTADOR A CERO
2991 61C9 9C 4E 03 STY C+1
3001 61CC F8 37 SED HV7 JDIFFERENCIAH ABSOLUTA

```

J  
\*\*\* DUCLE PRINCIPAL DE DIBUJO \*\*\*  
J

```

3041 61D0 4E 4F 03 HV5 LDX I JOSTENER DIRECCION EN X
3051 61D3 18 CLC JCALC, E=E+DELTY
3061 61D4 60 49 03 LDR E
3071 61D7 6D 47 03 SBC DELTY
3081 61DA 60 49 03 STR E JPRIMER BYTE BAJO
3091 61DD 60 4A 03 LDR E+1
3101 61E0 6D 48 03 ROR DELTY+1
3111 61E3 6C 4A 03 STX E+1
3121 61E6 30 14 BHI HV6
3131 61E9 28 SEC JCALC, E=E-DELTX
3141 61EB H0 49 03 LDR E
3151 61ED 6D 45 03 SBC DELTX
3161 61F0 6D 49 03 STR E
3171 61F3 6D 4A 03 LDR E+1
3181 61F6 6D 46 03 SBC DELTX+1
3191 61F9 6D 4A 03 STR E+1
3201 619B 08 BIC JX INC.
3211 6199 20 6A 01 HV6 JSR OUTPLT JSACAR UN MOVIMIENTO
3221 619C 6C 4D 03 INC I JINC CONTADOR
3231 619F 00 03 BNE HV7
3241 61A1 6E 4E 03 INC C+1

```

J  
JSAUTAR AQUÍ EN EL LER PASO  
J

```

3281 61A4 61 0D HV7 LDR JBYT.V JENCENDER UN PUNTO
3291 61A6 0C 44 03 ORA BITNO
3301 61A9 91 0D STR JBYT.V
3311 61AB H0 4D 03 LDR C JHECHO CUANDO C = DELTX
3321 61AE 0D 45 03 CMP DELTX
3331 61B1 H0 4E 03 LDR C+1
3341 61B4 6D 46 03 SBC DELTX+1
3351 61B7 98 04 BCC HV5 JTERMINADO
3361 61B9 68 RTS

```





```

LDN DIR. HEXADEC. ETIQUETA COD OPER ;COM. PRG: "ALTA RESOLUCION"
4151 6224 00 00 BNE UP3 ;VALID INICIALCIONAL
4161 6226 00 00 UP1 LDA BYT ;DECREMENTAR
4171 6228 00 02 BNE UP2
4181 622A C6 FE DEC BYT+1
4191 622C C6 FD UP2 DEC BYT
4201 622E 00 UP3 RTS

4211 622F 20 11 62 UL JSR UP ;PRIMERO ARRIBA, LUEGO IZO.
4221 6232 0C 44 00 LEFT ASL BITH0 ;1 PUNTO IZO.
4231 6235 00 00 SCC LF2 ;SI CORRECCION EN CARRY 0
4241 6237 2C 44 00 ROL BITH0 ;BITH0+1 Y CARRY+0
4251 6239 05 FD LDA BYT
4261 623C E9 07 SBC #7 ;-8 PORQUE CARRY=0
4271 623E 05 FD STA BYT
4281 6240 00 02 RCL LF2
4291 6242 C6 FE DEC BYT+1
4301 6244 00 LF2 RTS

4311 6245 20 FD 01 LR JSR DOWN ;PRIMERO ABAJO LUEGO DER
4321 6248 4E 44 00 RIGHT LSR BITH0 ;1 PUNTO DERECHA
4331 624B 00 00 RCL RTS
4341 624D C6 44 00 ROR BITH0 ;BITH0+00BY CARRY=0
4351 6250 05 FD LDA BYT
4361 6252 09 00 ROL #0
4371 6254 05 FD STA BYT

4381 6256 00 02 SCC RTS
4391 6258 C6 FE INC BYT+1
4401 625A 00 RTS

;
;***** CLRR *****
;
;IGORA EXACTAMENTE 8000 BYTES
;DEJA Y=000
;
4491 625B 00 3F CLRR LDA #0HLAST
4501 625D 05 FE STA BYT+1 ;INIC. PUNTERO A ULT PAGINA
4511 625F 05 00 LDA #0
4521 6261 05 FD STA BYT
4531 6263 00 ;
4541 6264 05 FD STA BYT
4551 6266 01 FD STA (BYT),Y
4561 6268 00 3F LDY #0HLAST ;EMPEZAR EN BASE+010F
4571 626A 02 20 LDH #020 ;EN X SE CUENTAN LAS PAGINAS
4581 626C 01 FD CLRR1 STA (BYT),Y ;ESCRIBIR CEROS
4591 626E 00 DEY
4601 626F 00 FD BNE CLRR1
4611 6271 C6 FE DEC BYT+1
4621 6273 C6 DEY
4631 6274 D8 F6 BNE CLRR1 ;32 PAGINAS
4641 6276 00 RTS

;
;***** SETCOL *****
;
;ESTABLECER COLORES DE TRAZO Y FONDO
;
4701 6277 09 50 SETCOL LDA #COLOR #EN 1 BYTE
4711 6279 02 00 SETCOL LDH #0
4721 627B 90 00 04 SETCL1 STA SCREEN,X ;4 PAGINAS
4731 627E 90 00 05 STA SCREEN+0000,X
4741 6281 90 00 0C STA SCREEN+00200,X
4751 6284 E0 INX
4761 6285 D8 F4 BNE SETCL1
4771 6287 AC E0 LDH #SCREEN+01 ;LA ULTIMA PAGINA
4781 6289 90 FF 06 SETCL2 STA SCREEN+002FF,X
4791 628C C6 DEY
4801 628D 00 FA BNE SETCL2
4811 628F 40 RTS

;
;PRINIT - ESTABLECE MODO ALTA RESOLUCION
;
4851 6290 00 11 00 PRINIT LDA #CTRL ;MODO ALTA RES.
4861 6293 09 20 ORA #020 ;BIT 5 A 1
4871 6295 00 11 00 STA #CTRL
4881 6298 00 19 00 LDA #HWRD ;BYT EN #2000
4891 629A 09 00 ORA #000 ;BIT 3 A 1
4901 629C 00 19 00 STA #HWRD
4911 629E 20 77 62 JSR SETCOL ;FORZAR NEGRO Y VERDE

```

```

LDA 6280 RD 00 03      JSR CLAMP      ;TODO CEROS
492: 6280 RD 00 03      LDA ERVECC      ;VECTOR DE ERROR DEL SISTEMA
493: 6280 RD 00 03      STR ERVECC
494: 6280 RD 01 03      LDA ERVECC+1
495: 6280 RD 01 03      STR ERVECC+1
496: 6280 RD 01 03      LDA ERVECC+1
497: 6280 RD 01 03      STR ERVECC+1
498: 6280 RD 01 03      STR ERVECC+1
499: 6280 RD 01 03      STR ERVECC+1
500: 6280 RD 01 03      STR ERVECC+1
501: 6280 RD 01 03      STR ERVECC+1
502: 6280 RD 01 03      STR ERVECC+1
503: 6280 RD 01 03      STR ERVECC+1
504: 6280 RD 01 03      STR ERVECC+1
505: 6280 RD 01 03      STR ERVECC+1
506: 6280 RD 01 03      STR ERVECC+1
507: 6280 RD 01 03      STR ERVECC+1
508: 6280 RD 01 03      STR ERVECC+1
509: 6280 RD 01 03      STR ERVECC+1
510: 6280 RD 01 03      STR ERVECC+1
511: 6280 RD 01 03      STR ERVECC+1
512: 6280 RD 01 03      STR ERVECC+1
513: 6280 RD 01 03      STR ERVECC+1
514: 6280 RD 01 03      STR ERVECC+1
515: 6280 RD 01 03      STR ERVECC+1
516: 6280 RD 01 03      STR ERVECC+1
517: 6280 RD 01 03      STR ERVECC+1
518: 6280 RD 01 03      STR ERVECC+1
519: 6280 RD 01 03      STR ERVECC+1
520: 6280 RD 01 03      STR ERVECC+1
521: 6280 RD 01 03      STR ERVECC+1
522: 6280 RD 01 03      STR ERVECC+1
523: 6280 RD 01 03      STR ERVECC+1
524: 6280 RD 01 03      STR ERVECC+1
525: 6280 RD 01 03      STR ERVECC+1
526: 6280 RD 01 03      STR ERVECC+1
527: 6280 RD 01 03      STR ERVECC+1
528: 6280 RD 01 03      STR ERVECC+1
529: 6280 RD 01 03      STR ERVECC+1
530: 6280 RD 01 03      STR ERVECC+1
531: 6280 RD 01 03      STR ERVECC+1
532: 6280 RD 01 03      STR ERVECC+1
533: 6280 RD 01 03      STR ERVECC+1
534: 6280 RD 01 03      STR ERVECC+1
535: 6280 RD 01 03      STR ERVECC+1
536: 6280 RD 01 03      STR ERVECC+1
537: 6280 RD 01 03      STR ERVECC+1
538: 6280 RD 01 03      STR ERVECC+1
539: 6280 RD 01 03      STR ERVECC+1
540: 6280 RD 01 03      STR ERVECC+1
541: 6280 RD 01 03      STR ERVECC+1
542: 6280 RD 01 03      STR ERVECC+1
543: 6280 RD 01 03      STR ERVECC+1
544: 6280 RD 01 03      STR ERVECC+1
545: 6280 RD 01 03      STR ERVECC+1
546: 6280 RD 01 03      STR ERVECC+1
547: 6280 RD 01 03      STR ERVECC+1
548: 6280 RD 01 03      STR ERVECC+1
549: 6280 RD 01 03      STR ERVECC+1
550: 6280 RD 01 03      STR ERVECC+1
551: 6280 RD 01 03      STR ERVECC+1
552: 6280 RD 01 03      STR ERVECC+1
553: 6280 RD 01 03      STR ERVECC+1
554: 6280 RD 01 03      STR ERVECC+1
555: 6280 RD 01 03      STR ERVECC+1
556: 6280 RD 01 03      STR ERVECC+1
557: 6280 RD 01 03      STR ERVECC+1
558: 6280 RD 01 03      STR ERVECC+1
559: 6280 RD 01 03      STR ERVECC+1
560: 6280 RD 01 03      STR ERVECC+1
561: 6280 RD 01 03      STR ERVECC+1
562: 6280 RD 01 03      STR ERVECC+1
563: 6280 RD 01 03      STR ERVECC+1
564: 6280 RD 01 03      STR ERVECC+1
565: 6280 RD 01 03      STR ERVECC+1
566: 6280 RD 01 03      STR ERVECC+1

```



UTH DEMO: HMMODEL, ETIOQUETA: CUD:R&R JCOM, PRO: "ALTA RESOLUCION"

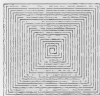
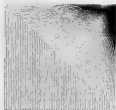
```

R20: 4000 EP 63      ,HMMODEM=1
R21: 4020 44 62      ,HMMODEM=1
T20: 4020 EP 63      ,HMMODEM=1
S20: 4020 EP 63      ,HMMODEM=1

S21: 4000 40 40 20 05000 ,BYTEND 400,420,430
S22: 4020 00 04 82      ,BYTEND 404,404,404
S24: 4000 01
S25: 4000
    
```



*Muestra de algunos de los  
gráficos que se pueden  
realizar con este  
programa*





# Telegrafía con ordenadores

No estaría completa una serie de programas para los radioaficionados si no incluyera un programa que permitiese practicar tanto la emisión como la recepción de código morse con la ayuda de nuestros ordenadores personales.

**H**e recibido gran cantidad de cartas a raíz de la publicación del primer artículo de esta pequeña serie, y en primer lugar debe pedir disculpas a todos por no poder contestar inmediatamente a cada uno, espero que lo comprendáis y seréis pacientes esperando la respuesta. Como es humano y no puedo hacerlo todo a la vez.

Para que sirva de adelanto en la respuesta a muchas de las cartas, contestaré a las preguntas más repetidas.

1. Intentaré preparar los esquemas de los modelos de R11V y C.W. tan pronto como sea posible, pero, como muchos imaginarán, al diseñarlos, hice un esquema teórico que, tras muchas modificaciones y pruebas, en nada se pareció al definitivo (supongo que los aficionados a realizar sus circuitos sabrán que esto es normal).

2. El dibujo de la placa de circuito impreso no lo tengo en blanco, ya que cumpliendo los puntos

lados de la ley de Murphy olvide los taladros y postes que unían varios componentes, y a consecuencia los "componentes flotantes" que fueron añadidos al dibujo original cuando los radioaficionados comenzaron a confirmar los postulados de Murphy que dicen aquello de que los circuitos no funcionan y sus amplificadores y filtros autooscilan.

3. De los dos puntos anteriores se puede deducir que actualmente no dispongo del esquema teórico o práctico definitivo para poder publicarlo. Lo siento lo puedo obtener un fin de semana de estos si desearan mis modelos y sus anotando los cambios realizados. Luego seré quedará pasando a un dibujo limpio y contrastado para imprimir.

Para no ocupar la misma con el esquema práctico, ya que dice de decirme cuántos impresos hace tiempo y se perdió el tranquilo que eso no puede prometer que lo publicaré, pero intentaré hacerlo dentro de un par de meses o tres, no me voy a poner a hacer.

4. Muchos han preguntado como adaptar el programa de R11V para telegrafía. Para todos estos, la respuesta está en este número, es otro programa distinto que permite emitir y recibir C.W. Quien dispone de tiempo y lo quiere montar con C.W. puede hacerlo a la manera, pero creo que es preferible tenerlos separados, ya que de este modo conseguirá más rapidez de construcción, además este programa altera el contenido de varios de los registros del port de usuario, y esto podría dar lugar a dificultades con el programa de R11V.

Espero que esto sirva de adelanto para la mayoría de los colegas que han escrito, obviamente me estoy intentando por carta pero a poco a poco para no mantenernos en suspense, pero a oscuras como la oscuridad el programa y cómo utilizarlo.

Las líneas 1 a 10 muestran los valores del crap de sonido y del port de usuario.

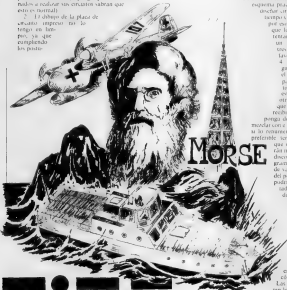


Illustration by



no para adaptarlo a nuestras necesidades. En los comentarios del programa podrás ver lo que hace cada una, y si quieres, puedes cambiar la nota que se escucha en el altavoz de TV cambiando el valor de la variable NO [nota] en la línea 9.

Las líneas 11 y 12 incrustan las dos matrices con los códigos de emisión (TXS) y recepción (RXS). Los datos están contenidos en las líneas 34 y siguientes.

Las líneas 13 a 30 señalan cuáles son las teclas a las que se les han asignado códigos especiales y emiten el indicativo y los mensajes asignados a las teclas de función.

La línea 31 fija la longitud de un punto en emisión (LPI) según la velocidad escogida.

Las líneas 32 a 38 son la rutina de recepción de caracteres (en los REM del programa cuenta lo que hace cada una), y según el algoritmo típico de simular dos registradores de desplazamiento de entrada de serie/valida paralelo. Este es el procedimiento habitual cuando todo se hace por medio de hardware, pero es más elegante hacerlo por software si el ordenador no tiene disponibles este tipo de entradas.

Hablando en términos técnicos de hardware, a cada impulso de reloj se desplazan los dos registradores en serie, y el dato aportado a sus entradas es el inverso uno respecto a otro, será uno uno en el registro serie de los puntos si el tiempo que ha estado pulsado el manipulador es inferior al periodo del reloj, y será uno uno en el registro serie de las rayas si el tiempo fue mayor. El periodo de la señal de reloj se adapta en función de la duración de los signos recibidos, de este modo permite que la velocidad en recepción se regule automáticamente en función de la señal recibida, y finalmente el carácter recibido se calcula multiplicando por dos el contenido de la salida paralelo de rayas y sumándole el de puntos.

SE NO HABRIS ENTENDIDO NADA DEL PARRAFO ANTERIOR NO DEBES PREOCUPARTE, el programa funciona bien aunque no sepas cómo lo hace, los detalles van destinados a aquella minora de élite que domina la electrónica digital y la informática, y quieren saber cómo y por qué un programa funciona de tal o cual modo.

En las líneas 31, 37 y 40 he añadido unas sentencias POKE a la memoria de pantalla, lo hace para facilitar mi propia práctica con el "machibabébre", viendo el contenido de las variables que cuentan la duración de un punto de una raya y el modo de tiempo por modo del carácter que aparece en cada una de las posiciones. Tampoco tienen que hacerlos, machibabé es sólo (podéis quitarlo si queréis). Los que tengan el kernel version 2 (lanjuno) no verán los caracteres y no hacen el correspondiente POKE a la memoria de color en la versión 03 lo hace el programa.

Los que tengan el VIC deberán cambiar estas POKES a la posición que ocupe su pantalla (depende de RAM de que dispongan), o, si prefieren, pueden quitarlos.

Las líneas 36 a 79 son la rutina de emisión, y otro que no tiene ningún misterio.

La línea 61 comprueba si se ha pulsado una de las teclas de función (mensajes programados).

La línea 62 pasa a recepción si se pulsa la tecla R con la tecla SHIF.

La línea 63 comprueba si el símbolo es la flecha (H) y, en caso, si el di di di di di di (V A) pasando después a recepción.

La 64 hace una pausa de una duración equivalente a uno veces la del punto, esta es la separación de palabras.

La 65 cambia la velocidad de emisión si se pulsa la tecla V con la tecla SHIF.

La 67 comprueba si el carácter es un signo que este periodo y vuelve al inicio de emisión.

La rutina de las líneas 69 a 75 es la que emite cada carácter a por el usuario haciendo al mismo tiempo que lo escuche en el altavoz de TV.

La rutina de la línea 76 es la que se encarga de emitir los mensajes preprogramados (el CQ CQ de ... y los otros tres que se comentan).

La rutina de la línea 81 es la que introduce la velocidad de emisión, que debe ser de uno a cuarenta caracteres por minuto.

Las líneas DATA de la 84 a la 87 contienen los datos de la matriz de recepción (RXS), y de la 88 a la 91 los de emisión (TXS).

Si quieres asignar funciones diferentes a las teclas podrás hacerlo cambiando estas líneas.

\*\*\*\* MUCHO CUIDADO CON OLVIDAR NINGUNA COMA, UNO, CERO O TRES, un solo error puede hacer que la letra Z ← → suene como G ← → o algo similar.

Aunque el programa funciona perfectamente en el VIC-20, prede que en el C-64 de problemas, ya que es bastante más lento (aproximadamente un 20 %). Estos problemas no deben apreciarse mientras las emisiones que necesitamos emitan con una cadencia regular y con los signos guardando la relación correcta (3/1), pero si aparecerán si la manipulación del correspondiente es deficiente y además ante a más de 18 palabras por minuto (mal emitido ya que al no sabe guardar una cadencia regular deteriora todavía su velocidad). Evite incóvenientes puede solventarse haciendo que el tamaño contados de recepción funcione más rápido para tener más presión en la detección (al ver mayores las otras adelantadas por los comandos del error es menor). Existen varios modos de conseguir mayor velocidad, uno de ellos es pasar el programa a código máquina, pero mi yo mismo me he entretenido en hacerlo, ya que la otra solución es mucho más sencilla y sencilla. COMPILE el programa con cualquiera de los compiladores que están disponibles en el mercado nacional (Blitz, Pet-Speed, Auto-Compiler, Benson's Compiler, etc.). Yo he utilizado el primero de ellos y la velocidad de ejecución del programa de recepción se ha visto incrementada unos 10 o 20 veces, con ello permite alcanzar velocidades de recepción de unos 50 caracteres por minuto con la precisión que antes recibía a 5 o 6 caracteres por minuto.

Tanto el VIC como el 64 no recibirán correctamente el código morse si la manipulación es deficiente, ya que aunque el margen de tolerancia en la duración de los signos es bastante amplio, todo tiene un límite (aunque se cumple el programa).

Un detalle importante que se me olvidaba: debemos alinear la constante que dividimos por la velocidad para calcular la duración de un punto, es el número 700 que aparece en las líneas 31 y 85, ya que si no lo hacemos, el programa emite mucho más rápido de lo que debe hacerlo (el número debes calcularlo por el procedimiento de tamaño, compile y prueba).

El listado que aparece completo es para el C-64, para el VIC-20 sólo publicamos las líneas que deben cambiarse, ya que el resto del listado es el mismo, aunque conviene que lo quiten todos los REM al programa para aumentar su velocidad y reducir el tamaño.

Un detalle importante que se me olvidaba: debemos alinear la constante que dividimos por la velocidad para calcular la duración de un punto, es el número 700 que aparece en las líneas 31 y 85, ya que si no lo hacemos, el programa emite mucho más rápido de lo que debe hacerlo (el número debes calcularlo por el procedimiento de tamaño, compile y prueba).

Finalizar el programa y si ya tienes algún modelo para CW, suerte en la cara del DX.

E-444E QRY

#### LISTADO PARA EL C-64

```
1 DR=36579:REM DIRECTION REGISTER C1
2
3 DR=36577:REM INPUT/OUT REGISTER C1
4
5 VO=34296:REM VOLUME SID
6 POKE34272,054296:POKE2,0:REM
5 POKE34296,15:REM VOLUME MAXIMO
6 POKE34277,14:REM ATTITUDE/CRUI
7 POKE34276,17:REM ONDA TRAFUGAL
8 POKE34278,128:REM SISTEMAS/SCHE
9 DE=54273:IND=48:REM GENERADOR/NOTA
10 PRINT"ELC13":POKEOR,128:POKEOR,0:
```

POKE5089,63

```

11 DIMA$(100),T$(60)
12 FORC=1TO100:RNDX=(Z)+NEXT(4082+
13 T$(60)+RNDT$(42)+NEXT
14 PRINT"CLASIFICACION"SPC3TECLAS(SPC3
15 ESPC3ALES)
16 PRINT"ESPC3A=COMPRESION(SPC3)C=0"
17 PRINT"ESPC3B=FH+HEMSAJE(SPC3)R"
18 PRINT"ESPC3C=ERROR"
19 PRINT"ESPC3D=OK"
20 PRINT"ESPC3E=FH+EMISION(SPC3)R"
21 PRINT
22 PRINT
23 PRINT
24 PRINT"PULSA(SPC3)HERR(SPC3)TECLA"
25 GETA:IF A="" THEN G25
26 INPUT"CLASIFICACION"SPC3TUS(SPC3)H
27 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
28 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
29 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
30 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
31 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
32 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
33 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
34 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
35 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
36 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
37 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
38 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
39 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
40 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
41 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
42 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
43 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
44 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
45 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
46 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
47 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
48 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
49 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
50 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
51 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
52 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
53 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
54 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
55 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
56 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
57 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
58 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
59 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
60 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
61 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
62 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
63 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
64 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
65 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
66 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
67 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
68 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
69 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
70 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
71 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
72 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
73 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
74 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
75 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
76 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
77 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
78 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
79 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
80 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
81 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
82 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
83 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
84 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
85 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
86 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
87 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
88 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
89 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
90 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
91 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
92 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
93 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
94 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
95 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
96 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
97 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
98 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
99 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)
100 INPUT"CLASIFICACION"SPC3HEMSAJE(SPC3)

```

```

11 IFORND1 THENG25:REM ESPERA 100MS, SU
12 EL TO
13 POKEDE,0:REM 2000+PU=2000:IF C=0/2
14 THENG25:PU=1:IF C=1 THENG25:1/5:GOTO43
15 IF C=0/2 THENG25:REM 1:IF C=1 THENG25
16 L=0
17 L=0
18 G25:PEEK(10) L=L+1:IFORND1 THENG25
19 IF C=0 THENG25
20 IF C=1 THENG25
21 IF C=2 THENG25
22 IF C=3 THENG25
23 IF C=4 THENG25
24 IF C=5 THENG25
25 IF C=6 THENG25
26 IF C=7 THENG25
27 IF C=8 THENG25
28 IF C=9 THENG25
29 IF C=10 THENG25
30 IF C=11 THENG25
31 IF C=12 THENG25
32 IF C=13 THENG25
33 IF C=14 THENG25
34 IF C=15 THENG25
35 IF C=16 THENG25
36 IF C=17 THENG25
37 IF C=18 THENG25
38 IF C=19 THENG25
39 IF C=20 THENG25
40 IF C=21 THENG25
41 IF C=22 THENG25
42 IF C=23 THENG25
43 IF C=24 THENG25
44 IF C=25 THENG25
45 IF C=26 THENG25
46 IF C=27 THENG25
47 IF C=28 THENG25
48 IF C=29 THENG25
49 IF C=30 THENG25
50 IF C=31 THENG25
51 IF C=32 THENG25
52 IF C=33 THENG25
53 IF C=34 THENG25
54 IF C=35 THENG25
55 IF C=36 THENG25
56 IF C=37 THENG25
57 IF C=38 THENG25
58 IF C=39 THENG25
59 IF C=40 THENG25
60 IF C=41 THENG25
61 IF C=42 THENG25
62 IF C=43 THENG25
63 IF C=44 THENG25
64 IF C=45 THENG25
65 IF C=46 THENG25
66 IF C=47 THENG25
67 IF C=48 THENG25
68 IF C=49 THENG25
69 IF C=50 THENG25
70 IF C=51 THENG25
71 IF C=52 THENG25
72 IF C=53 THENG25
73 IF C=54 THENG25
74 IF C=55 THENG25
75 IF C=56 THENG25
76 IF C=57 THENG25
77 IF C=58 THENG25
78 IF C=59 THENG25
79 IF C=60 THENG25
80 IF C=61 THENG25
81 IF C=62 THENG25
82 IF C=63 THENG25
83 IF C=64 THENG25
84 IF C=65 THENG25
85 IF C=66 THENG25
86 IF C=67 THENG25
87 IF C=68 THENG25
88 IF C=69 THENG25
89 IF C=70 THENG25
90 IF C=71 THENG25
91 IF C=72 THENG25
92 IF C=73 THENG25
93 IF C=74 THENG25
94 IF C=75 THENG25
95 IF C=76 THENG25
96 IF C=77 THENG25
97 IF C=78 THENG25
98 IF C=79 THENG25
99 IF C=80 THENG25
100 IF C=81 THENG25

```



# BOXER 12

high resolution monochrome monitor 12"

NEW 85  
NOVEDAD 85

## ELECTRICAL ENVIRONMENTAL CHARACTERISTICS

CRT	SIZE	12"
	DEFL. ANGLE	25°
DISPLAY FORMAT	CHARACTERS	2000 (80 x 25)
VIDEO	VIDEO SIGNAL	COMPOSITE VIDEO
	VIDEO SIGNAL	1 Volt p-p
	RISE/FALL TIME	< 30 ns
	BANDWIDTH	20 MHz
	CENTRE RESOLUTION LINES/cm	1500
	INPUT RESISTANCE	15 Ohm
BLANKING TIME	HORIZONTAL	< 8 µs
	VERTICAL	< 700 µs
COMP. SYNC	H. SYNC	15.625 - 15.750 kHz
	V. SYNC	50 - 60 Hz
SH	[E - C]	12 EV
POWER SUPPLY	INPUT VOLTAGE	100 - 120 VAC 50/60 Hz
	CONSUMPTION	30 VA
GEOMETRY	RASTER DISTORTION	max 1 %
	SCAN LINEARITY	max 15 %
	FOCUS	external control
	V. AMPLITUDE	external control
	V. FREQUENCY	external control
	V. UPPER AND LOWER LINEARITY	external control
	H. AMPLITUDE	external control
	H. FREQUENCY	external control
	H. LINEARITY	external control
	H. PHASE	external control
ENVIRONMENTAL	AMBIENT TEMPERATURE	21° C - 40° C
	AMBIENT HUMIDITY (not condensed)	5-80 %
	STORAGE TEMPERATURE	40° C - 60° C
	STORAGE HUMIDITY (not condensed)	5-80 %
WEIGHT	GROSS-Net	5.750 Kg

■ audio optional

## HANTAREX

NO  
POWER

BOXER 12



ON / OFF

ON / OFF

TRATTAMENTO SCHERMO: SCURO - ANTIRIFLETTENTE  
SCREEN TREATMENT: DARK GLASS - ETCHED

FOSFORO - P31 - VERDE MEDIO-BREVE  
PHOSPHOR - P31 - GREEN MEDIUM-SHORT

DATI MECCANICI  
MECHANICAL DATA



**HANTAREX**<sup>®</sup>  
QUALITY. RELIABILITY. SERVICE

Electronic  
Equipment  
Manufacturer

Aragón, 210, 1°, 1ª - Barcelona 11 - teléf. (93) 3232041 - telex 98017

PARA EL VIC-20  
BORRAR LAS LINEAS 1 A 10  
Y SUSTITUIRLAS POR:

```
1 OR=27128
2 IR=27136
3 POK=36870,15:REN VOLUMEN MAXIMO
4 POK=36874,0
5 SE=36874:HS=245:REN GENERADOR/MOTR
6
10 PRINT"CLL3"POKEIR,120:POKEIR,0:
POKE37140,124
33 DR=PEEK(1R):DFORWARD:THEN36
37 DR=PEEK(1R):IL=L+1:IF(DR=0):GOTO43
43
48 DR=PEEK(1R):IL=L+1:IF(DR=0):THEN39:
REN ESPERA RENDEP,SUETO
```

SUMA DE CONTROL DEL PROGRAMA MORSE-64

1	84	2	109	3	112
4	138	5	45	6	92
7	87	8	13	9	17
10	221	11	233	12	227
13	143	14	163	15	35
16	243	17	113	18	230
19	255	20	168	21	153
22	153	23	153	24	239
25	52	26	179	27	87

28	92	29	97	30	6
31	99	32	68	33	172
34	117	35	111	36	169
37	8	38	129	39	116
40	38	41	162	42	90
43	46	44	162	45	238
46	192	47	18	48	54
49	58	50	53	51	61
52	57	53	82	54	177
55	177	56	126	57	38
58	97	59	121	60	51
61	98	62	82	63	232
64	30	65	95	66	89
67	186	68	188	69	84
70	97	71	196	72	31
73	5	74	137	75	230
76	38	77	140	78	137
79	225	80	70	81	235
82	237	83	142	84	31
85	68	86	95	87	191
88	46	89	211	90	157
91	151	92	166	93	162

SUMA DE CONTROL DE LOS CAMBIOS DEL VIC-20

1	78	2	81	3	51
4	255	5	79	10	1
33	31	37	173	40	281

(Fin de pág. 3)

```
250
SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
600 NEXT I,RR
270 FOR T=1 TO 2500: NEXT T
300 SYS(1R)
210 D=4:IF(2:100:100)V=VC
320 SYS(1R),X,V
230 FOR T=0 TO 20
240 D=D+E:IF(V=0:SYS(1R),X,V
350 D=D+E:100:100:SYS(1R),X,V
360 D=D+E:IF(V=0:SYS(1R),X,V
370 D=D+E:100:100:SYS(1R),X,V
380 NEXT T
390 FOR T=1 TO 2500: NEXT T
400 SYS(1R),X=4PI:1/3
410 FOR T=0 TO 15:STEP 5
420 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
430 FOR T=1 TO 24:STEP 1
440 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
450 NEXT T
460 FOR T=1 TO 2500: NEXT T
500 SYS(1R),X=4PI:1/4:D=R/20
510 FOR T=0 TO 15:STEP 5
520 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
530 FOR T=1 TO 24:STEP 1
540 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
```

PROGRAMA "RETA RESOLUCION "

```
550 NEXT I
560 R=R-D: NEXT T
580 FOR T=1 TO 2500: NEXT T
600 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
610 FOR T=0 TO 15:STEP 5
620 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
630 FOR T=1 TO 24:STEP 1
640 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
```

```
650 NEXT I
660 R=R-D: NEXT T
680 FOR T=1 TO 2500: NEXT T
700 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
710 FOR T=0 TO 15:STEP 5
720 T=R/100
730 X=1:JC=RND(1/24:100:100),VC=RND(1/24:100:100)
740 NEXT T
750 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
760 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
770 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
780 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
790 FOR T=1 TO 2500: NEXT T
800
810 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
820 FOR T=1 TO 24:STEP 1
830 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
840 NEXT T
850 FOR T=1 TO 2500: NEXT T
860 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
870 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
880 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
890 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
900 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
910 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
920 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
930 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
940 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
950 NEXT T
960 FOR T=1 TO 2500: NEXT T
970 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
980 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
990 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
1000 SYS(1R),JC=RND(1/24:100:100),VC=RND(1/24:100:100)
1010 NEXT T
1020 GET:IF(1/24:100:100) THEN 820
9999 FOR T=1 TO 2500: NEXT T
9999 SYS(1R)
```

algunos de nuestros programas para los  
ordenadores personales

**C<sub>64</sub> commodore 64**



**¡Aumente la capacidad  
de su C<sub>64</sub> commodore 64  
hasta 1.025 Kb.**



MOD.1541:170Kb.  
MOD.1001:1025Kb.

**B.M.**

**BASIC MICRO-ORDENADORES, S.A.**

AVD. CESAR AUGUSTO,72  
Tifs. 23 56 82 y 22 65 44  
50003 ZARAGOZA

SE NECESITAN DISTRIBUIDORES

se adaptan





# Su Commodore 64 tiene mucho que decirle. Unidad de Disco.

El Commodore 64 es el resultado de la experiencia internacional de Commodore como líder indiscutible en el mercado de los microordenadores.

El Commodore 64 es el ordenador más completo y potente de su categoría, ... pero todavía tiene mucho que decirle.

Por ejemplo su Unidad de Disco.

Sienta como aumenta notablemente la capacidad de memoria de su C-64, como agiliza la carga y descarga de programas y facilita la localización, casi instantánea, de cualquier dato.

Amplie las posibilidades de su C-64, descargando su extensa gama de periféricos.

Ahora que ya sabe que su Commodore 64 tiene todavía mucho que decirle, prepárese a conocerle mejor.

#### PRINCIPALES CARACTERISTICAS

- 170 K de capacidad - Ficheros secuenciales y relativos y de acceso directo - Unidad inteligente, con sistema operativo incorporada.

commodore 64

  
commodore



Microelectrónica y Control c/ Valencia, 49-53 08015 Barcelona - c/ Princesa, 47 3.º G 28008 Madrid  
Único representante de Commodore en España.